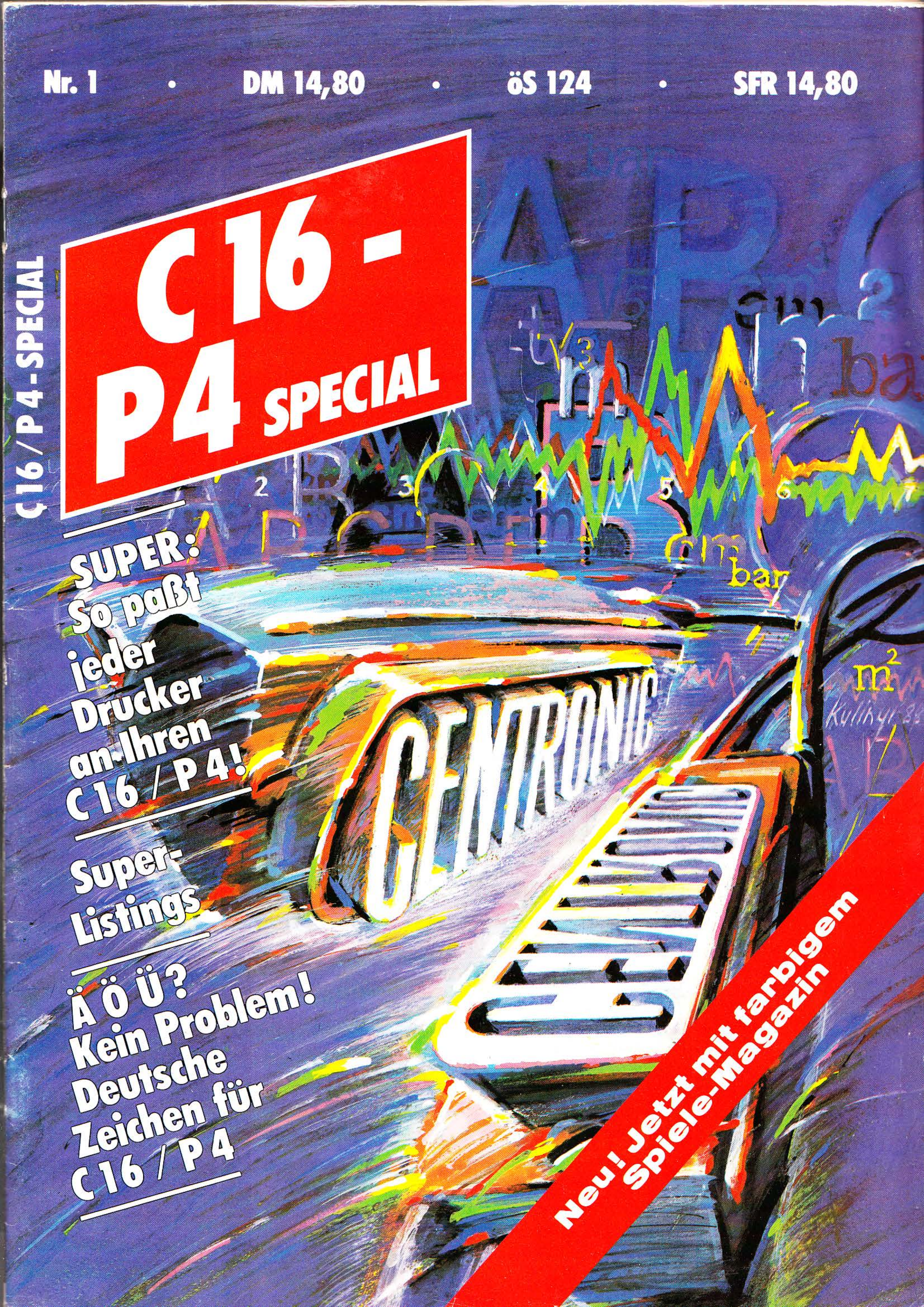

Nr. 1 • DM 14,80 • öS 124 • SFR 14,80
C 16 / P 4-SPECIAL
C 16 -
P 4 SPECIAL
SUPER: So paßt jeder Drucker an Ihren C 16 / P 4!
Super-Listings
Ä Ö Ü? Kein Problem! Deutsche Zeichen für C 16 / P 4
CENTRONIC
Neu! Jetzt mit farbigem Spiele-Magazin

DIE GEWINNER

Jetzt stehen sie fest, die Gewinner aus unserer Fragebogenaktion 1987/88.
Fortuna hat unsere Leserin Heike Merschbrok aus dem badischen Vörstetten für den ersten Preis auserwählt. Ihr Wunsch nach einem Commodore PC1 mit Bildschirm und Zubehör geht somit in Erfüllung. Sie wird den Computer in den nächsten Tagen in der Redaktion der COMMODORE WELT in München überreicht bekommen.



Das Echo unserer Fragebogenaktion war so enorm, daß eine zusätzliche Arbeitskraft eingestellt werden mußte, um den Posteingang überhaupt zu bewältigen. Auch die auswertende Crew wurde von dem Tatendrang unserer Leser überrascht. Wir danken Allen, die sich die Mühe gemacht haben, den Fragebogen an uns zurückzusenden. Sie geben uns so die Möglichkeit, die COMMODORE WELT an Ihre Wünsche noch besser anzupassen.
Alle anderen Gewinner wurden bereits benachrichtigt.

Ihre C16/P4-Crew

COMPUTER IN DER FORSCHUNG

Kollege Chip kommt gleich

Computer greifen in der Arbeitswelt immer mehr um sich. Die Rede ist vom Jobkiller. Das Elektronenhirn kann auch hilfreicher Kollege sein. Auch wenn man auf Schritt und Tritt die Rechner sieht, im Supermarkt oder in der Arztpraxis: Was genau damit gemacht wird, weiß nur der Eingeweihte. Das Bild von der Chip-Revolution, vor der sich jeder so fürchtet, stimmt nicht mehr.



Günter Strobl (links) und Gert Seidel im Gespräch

Für Computeranwender, die wissen wollen, ob sie ihre Computer-Erfahrung auch im Arbeitsleben gebrauchen können, ist ein Blick über den Zaun der eigenen Umgebung sinnvoll. Steht es dafür, seine Zeit mit Pascal und C zu vertun, oder werden die Rechner in den Betrieben nicht ohnehin mit einer Software ausgestattet, die den Benutzer dazu degradiert, stupide auf Knöpfe zu drücken? Bleibt vom einmal Erlernten nur das Wissen, was ein Cursor und eine Return-Taste ist? Dies waren Fragen, die ich mit einem Besuch im Toxikologie-Labor der GSF (Gesellschaft für Strahlenforschung und Umwelt) in München klären wollte.

UMWELT— GRUND GENUG ZUR FORSCHUNG

Die GSF gilt als eine der größten und angesehensten Forschungsinstitute der Bundesrepublik. Die Sorge um die Umwelt steht im Vordergrund, egal, ob es aktuelle Probleme sind wie etwa der Reaktorunfall von Tschernobyl oder das AIDS-Virus, oder ob es längerfristige Aufgaben zu lösen gilt.
Die zunehmende Verschmutzung von Luft, Wasser und Boden stehen an erster Stelle. Da diese Umweltprobleme vorwiegend von Giften hervorgerufen werden, galt unser Besuch dem Labor für Toxikologie. Hier steht die Erforschung von Substanzen und der ihnen eigenen Wirkungen im Vordergrund.
Dabei ist es zunächst gar nicht so wichtig, ein Mittel gegen das Übel zu finden, manchmal ist dies ohnehin nicht möglich. Vorrang hat die Grundlagenforschung über den Stoff und die Festlegung von vertretbaren Höchstwerten. Mit anderen Worten: Wie weit ist letztlich der Mensch damit belastbar, oder verbietet sich der Umgang mit derartigen Substanzen von vornherein?
Erst dann kommt die Pharmakologie, die Arzneimittelkunde, zum Zuge. Kein finanzieller Ertrag also, sondern Erforschung und Kontrolle der Umwelt.

EDV — START MIT DER TEXTVERARBEITUNG

Welche Rolle spielt der Computer in der Forschung der GSF? Günter Strobl, Chemiker, gab Auskünft über das Labor und sein Forschungsprojekt. Die erste Überraschung war, daß das Rechenzentrum der GSF kaum eine Rolle spielt. Nicht etwa, weil die Anwendungen zu unwichtig wären, ganz im Gegenteil.
Der Trend hat sich innerhalb weniger Jahre gewandelt. Bislang war es ja so, daß jede Forschungsgemeinschaft auf ein gemeinsam benutztes Rechenzentrum zugriff. War es einmal eine allgemeine Vorstellung, daß eine Zentraleinheit mehrere Terminals versorgt, so haben sich die Ansprüche gewandelt. Glücklicherweise kam es noch nicht zur Endausbaustufe eines solchen Netzwerkes. Heute sind,nicht nur in der GSF, Individualrechner angesagt, eben das, was man mit der Wortschöpfung PC meinte.

Unser Blick auf die Ausstattung in den Labors fiel zuerst auf einige Wang-Rechner, die speziell für die Textverarbeitung vernetzt wurden. Arbeitsberichte, Doktorarbeiten und jede Korrespondenz wird mit diesem Verbund erledigt. Daß man trotzdem nicht in Saus und Braus lebt, beweist die Tatsache, daß für die fünf Terminals nur ein Drucker vorhanden ist.
Überhaupt sind Computer trotz ihrer Bedeutung relativ neue Kollegen in der Forschungsgemeinschaft. Erst vor einem Jahr haben die Wang-Geräte Gesellschaft bekommen. IBM heißt nun die Devise und als Superstar darf neuerdings ein Wyse AT mit 80386-Prozessor seinen Dienst verrichten. So glaubte ich jedenfalls, bis mich Günter Strobl auf den Denkfeher hinwies.

GETARNTE COMPUTER

Wie fast alle Außenstehenden habe auch ich mich von Äußerlichkeiten beeindrucken lassen. Ich saß lange Zeit neben einem gerade beschäftigten Rechner, ohne etwas davon zu ahnen. Es handelte sich um ein HPLC-Gerät und wurde von mir deshalb nicht als Rechner erkannt, weil die typische alphanumerische Tastatur fehlte. Zweckgebunden sorgt hier ein Standardprozessor für Steuerung und Auswertung. Ein fest installiertes Programm läßt nur die Eingabe einiger Parameter, etwa die Versuchszeit, zu. HPLC heißt dabei nichts anderes als Hochdruck-Flüssigkeits-Chromatograph (High Performance Liquid Chromatograph).

Grob gesagt geht es darum, Substanzen in geringster Menge sowohl quantitativ oder auch qualitativ zu bestimmen.
Soweit mein Laienverstand begriff, werden hierzu die Moleküle über eine bestimmte Wegstrekke in einem flüssigen Medium transportiert. Für die Auswertung gibt eine gezackte Linie Auskunft, die sich wie eine Gebirgssilhouette darstellt.
Jeder einzelne Berg, richtig als Peak bezeichnet, läßt sich in Höhe und Fläche berechnen. Entweder gibt das Diagramm bei bekannten Stoffen Hinweise auf die Konzentration der Substanz in Lösung, oder es läßt sich im Vergleich mit bekannten Standardsubstanzen ermitteln, um welchen Stoff es sich handelt.
Ein recht komplexes Aufgabenfeld für einen Rechner, auch wenn das Äußere der Tastatur täuscht.
So mancher Homecomputer sieht leistungsfähiger aus als das, was sich da unter einer billigen Folientastatur verbirgt.

Wer nun aber glaubt, das Ganze sei ebenso einfach zu bedienen wie ein Videorekorder, der sollte sich das Handbuch anschauen oder es auch nur abwiegen. Auch wenige Tasten können für viel Verwirrung sorgen, und die Liste dessen, was falsch gemacht werden kann, ist ellenlang.
Trotz meiner Anerkennung für derartige Arbeiten galt mein Interesse dem Wyse-AT, an dem Chemiker Strobl schon bei meiner Ankunft arbeitete. Da auch in den Forschungslabors das Geld nicht reichlich fließt, mußte für den leistungsfähigen Rechner zunächst ein konkreter Bedarf, eine Anwendung, nachgewiesen werden. Daran mangelt es allerdings nicht.

DREIDIMENSIONALES MOLEKÜLWUNDER IM AT

Der Chemiker brütet über molekulare Wechselwirkungen. Das Problem ist – wie immer – schnell beschrieben: Jedes Molekül hat ein bestimmtes Aussehen, eine plastische, dreidimensionale Symmetrie. Dabei sagen gleiche Bruttoformeln und gleiches Molekulargewicht noch lange nicht aus, daß auch die Strukturformel gleich ist.
Isomere nennt man derartige Täuschungen der Natur, die auf den ersten Blick zwar identisch scheinen, trotzdem aber andere chemische und physikalische Eigenschaften haben.
Es kommt also auf die Symmetrie an, und hier setzt die Arbeit von Günter Strobl ein. Sein Werkzeug: Ein Wyse-AT mit großer Rechenleistung und – eine Stereobrille.
Sie haben richtig gelesen, es handelt sich um das altbekannte Verfahren, mit dem ein dreidimensionaler Effekt erzeugt wird. Die Farben Rot und Grün werden getrennt auf den Bildschirm ausgegeben und knapp nebeneinander plaziert.
Eine entsprechend eingefärbte Brille kann das Auge, in erster Linie aber



Wenig Tasten und aufwendige Technik

Bei der Computeranwendung in der Forschung geht der Trend weg vom zentralen Terminal, hin zur individuellen Anwendung. Nicht vorhanden oder praktisch unbezahlbar ist die Software. Programmierer in der Forschung – ein Beruf mit Zukunft.

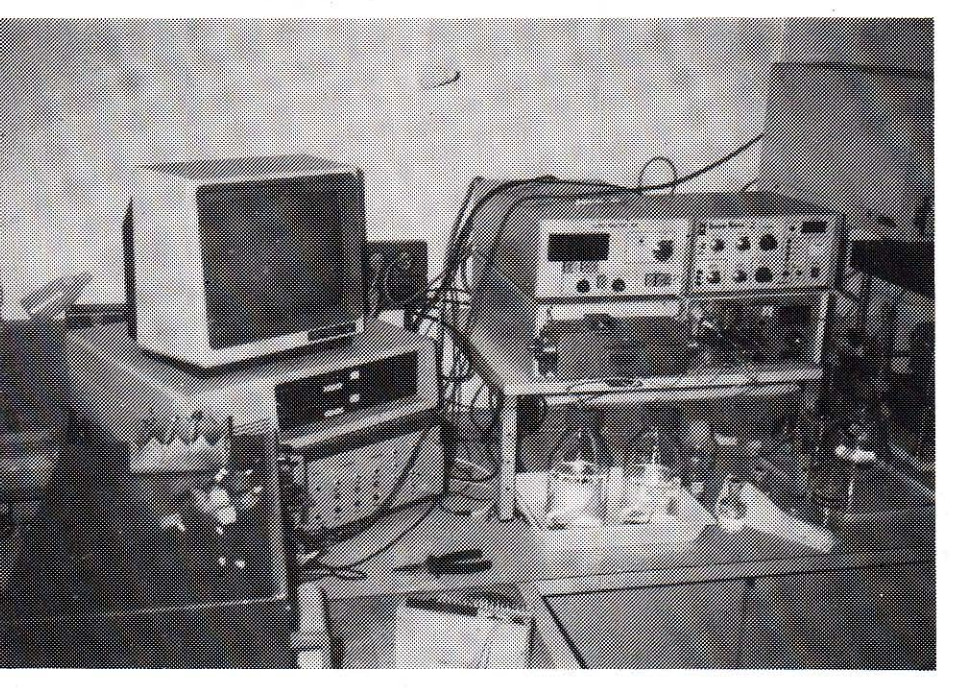
das Gehirn, täuschen und erzielt einen dreidimensionalen Eindruck.
Auf dem Sektor der Computerspiele gibt es einige derartige Versuche, und im Kino oder Männermagazinen haben Sie sicher schon Ähnliches gesehen. Falls es sich dabei um Bilder mit Jayne Mansfield handelte, wissen Sie, wie leicht sich das menschliche Gehirn von der Plastizität überzeugen läßt.
Ganz so vergnüglich geht es in der GSF allerdings nicht zu. Trotz guter Hardware lassen sich immer noch keine sehr großen Moleküle verarbeiten, aber vorerst gibt es auch mit kleineren Bildern genug zu tun. Aus der Erscheinungsform lassen sich Rückschlüsse auf die Wirkung der Substanzen ziehen.

LIGHTPEN IN DER KREBSFORSCHUNG

Recht unscheinbar gibt sich eine Computereinheit namens Videoplan. Hierbei geht es immerhin um Krebsforschung, eine der vordringlichsten wissenschaftlichen Aufgaben. Wichtigstes Werkzeug ist dabei ein Lightpen, der Computerzusatz, der bislang bei Homeanwendern eher mitleidig belächelt wurde.
Behandelte tierische Zellen werden in einem äußerst dünnen Querschnitt unter das Mikroskop gelegt und vom Computer abgetastet. Das auf dem Monitor entstehende Bild kann geprüft und bearbeitet werden.
Stößt der Wissenschaftler dabei auf anomale Zellen im Gewebe, so markiert er sie mit dem Lightpen und erhält über ein Rechenprogramm die exakten Werte. Zusätzlich werden sämtliche Zellen, die eine ähnliche Anomalie aufweisen, gezählt. Eine Statistik gibt Aufschluß, inwieweit das injizierte Material krebserregend war. Dazu muß vielleicht gesagt werden, daß eine einzige deformierte Zelle innerhalb eines Gewebes noch lange kein Alarmzeichen ist. Erst der Rechner, beziehungsweise das Programm gibt Aufschluß über den Grad der Krebsgefährdung.

GUTE AUSSICHTEN FÜR TIERFREUNDE

Dabei geht es natürlich nicht ohne Tierversuche. Kein angenehmes Thema. Es ist aber Tatsache, daß Tierversuche, abgesehen von denen der kosmetischen Industrie, notwendig sind. Deshalb finden sie auch im Bereich der Toxikologie der GSF statt, es geht letztlich um Menschenleben. Aber hier wirkt sich der Computer bereits segensreich aus.
Es gibt Datenbanken, in denen Basisversuche abgespeichert werden. Daß dieses Archiv genutzt wird, ist ein simples Rechenexempel, ist ist einfach billiger und schneller; Doktoranden müssen nicht immer wieder aufs neue toxische Werte im Tierversuch ermitteln, sondern können die Daten abfragen.
Auch rein theoretisch wird auf diesem Gebiet gearbeitet. Aufgrund vorhandener Ergebnisse simulieren Rechner den Ausgang eines neuen Versuches, ohne daß er wirklich durchgeführt wurde.



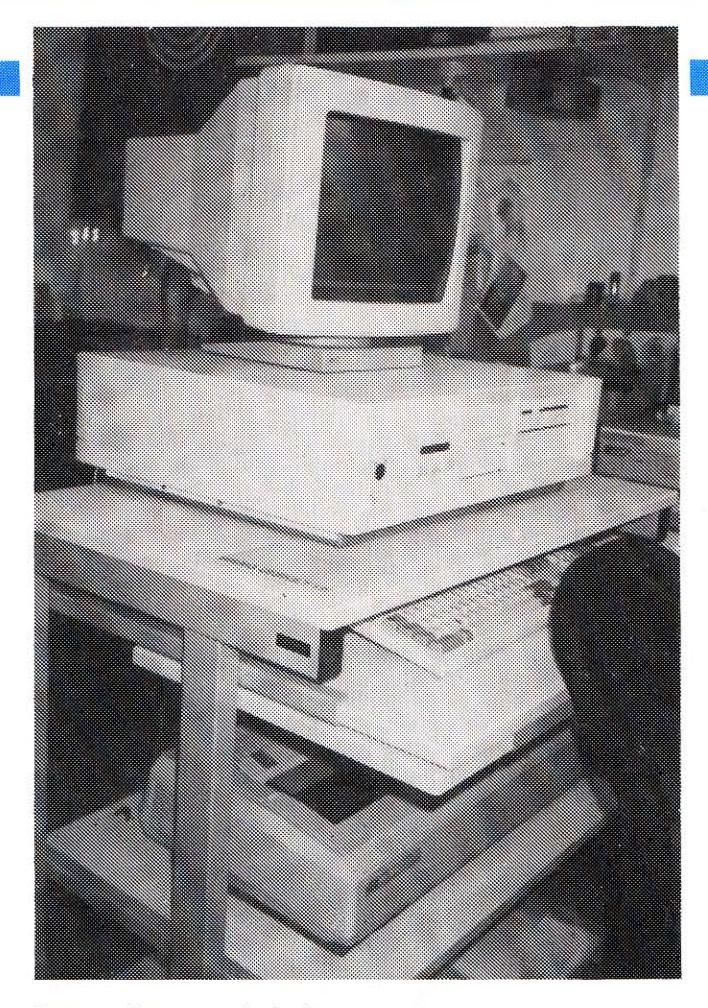
Ein IBM steuert den HPLC und übernimmt die Auswertung der Ergebnisse

SOFTWARE – WOHER NEHMEN UND NICHT STEHLEN?

Zumindest ich kannte die beschriebene Anwendung nur vom Hörensagen, und die naheliegende Frage war: Woher kommt die Software? Günter Strobl hat zwei Programme für diese Arbeit, wobei ein amerikanisches, relativ preiswertes, Angebot bereits in der Schublade verschwunden ist: nicht geeignet, kein effektiver Nutzen, bestenfalls einem Studenten zu empfehlen. Dem Mangel konnte mit einigen Beziehungen zur Technischen Universität und sehr viel Sachkenntnis abgeholfen werden.

UNBEZAHLBARE SOFTWARE

Ein persönlicher Bekannter stellte ein noch nicht ganz fertiges Programm zur Verfügung, mit dessen Leistungsfähigkeit man zufrieden ist. Hier ist Strobl plötzlich in der Rolle eines Pilotanwenders, eine Arbeit, die auch Informatikkenntnisse erfordert. Natürlich gibt es auch kommerzielle Programme. Der Kostenaufwand liegt allerdings bei einer Größenordnung von 60000 Mark, die Hardware in Form einer Workstation eingeschlossen. Zu viel.
Selbst dann, so befürchtet Günter Strob, sind die Anwendungen in der Wissenschaft zu speziell, und es dürfte schwierig werden, die geeignete Software zu finden. Abhilfe ist dort zu suchen, wo ebenfalls geforscht wird. Im Bereich Informatik der technischen Universitäten ist relativ viel zu holen. Nur befinden sich diese Programme fast alle noch im Entwicklungsstadium, werden selten vollständig veröffentlicht



Neueste Errungenschaft: Ein Wyse AT

und wenn überhaupt Dokumentation vorhanden ist, dann läßt diese sehr zu wünschen übrig. Vorteile hat diese Beschaffungsweise allerdings auch. Die Programme dürfen gar nicht kommerziell verwertet werden, sofern sie an der Hochschule entwickelt wurden. In den USA müssen mit öffentlichen Mitteln finanzierte EDV-Projekte sogar der Öffentlichkeit zugänglich gemacht werden.
Die Public-Domain-Szene ist Tummelplatz der besten Programme, mit denen Otto Normalverbraucher allerdings wenig anfangen kann. In der BRD sind wir noch nicht so weit, aber einige gute Kontakte helfen über die Hürden und bringen letztlich individuelle Software in den Computer, die unbezahlbar ist.
Aber dazu muß man fit auf dem Computer sein, denn abgesehen davon, daß die Software nicht ganz fertig und bis ins letzte Detail erprobt ist, stellt sich auch schnell heraus, daß der Programmierer kein Chemiker oder Biologe war. Wenn sich auch der Chemiker unfähig zur Programmverbesserung zeigt, kann ein ganzes Projekt ad acta gelegt werden.
Damit ist auch die Frage beantwortet, welche Aussichten Sie mit Ihrer in der Freizeit erworbenen EDV-Bildung haben. Verlangt wird sie (noch) nicht, wie es so schön heißt. Aber abgesehen davon, daß einmal erworbenes Wissen nie schadet, ist gerade der Forscher im naturwissenschaftlichen Bereich zunehmend auf Computeranwendung angewiesen. Und ohne Erfahrung muß der Selbstlehrgang eben nachgeholt werden. Da darf auch die Information nicht fehlen, mit welchen Programmiersprachen man es letztlich zu tun hat.
Bislang wird der Hauptteil der Software in Fortran eingegeben. FORmula TRANslation ist nun einmal die Sprache der Wissenschaftler. Gelegentlich greift man bei kleineren Routinen auf Assembler zurück.
Die große Trendwende steht jedoch bevor. Nach den Beobachtungen von Strobl wird zunehmend in C programmiert. Ein besonderer Aspekt, da man heutzutage bereits in Homecomputern die Möglichkeiten dieser Sprache nutzen kann.

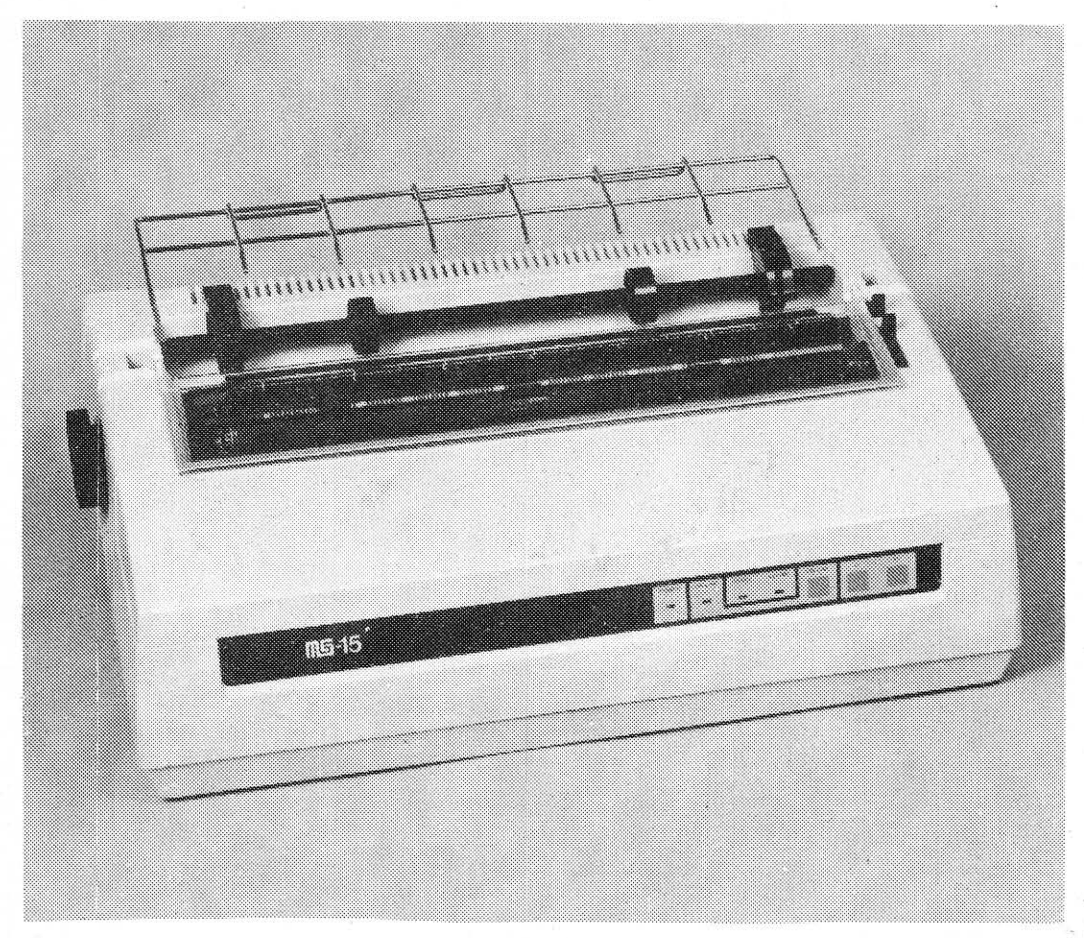
SPEKULATIONEN ZUR ZUKUNFT

Computer sind Vertreter einer zukünftigen Technik, die bereits heute schon erhältlich ist.
Was wird von seiten der Elektronik auf die Forschung zukommen?
In den Labors ist das Rennen um den Superrechner noch längst nicht gelaufen. Und es wird genau da entschieden und nicht etwa in den großen Rechenzentren. Die Forscher benötigen Rechner, die enorm leistungsfähig sind und trotzdem in einer Preiskategorie liegen, die es erlaubt, jedem einzelnen Wissenschaftler eine Konfiguration zur Verfügung zu stellen.
Um es noch einmal in Erinnerung zu bringen: Ein AT ist bereits heute mit der Darstellung von einfachen Molekülen ausgelastet. Komplexere Strukturen sind nur an einem größeren Rechner darzustellen, der wiederum von einem Wissenschaftler alleine an die Grenzen gebracht werden kann. In den Labors fehlt also nicht das gewaltige Netzwerk, sondern der Arbeitsplatz-Computer der Superlative.
Folgen hat dies natürlich auch auf die Ausbildung der Naturwissenschaftler. Falls Sie selbst ein Studium der Chemie, Medizin oder der Biologie planen und den Arbeitsbereich der Forschung im Auge haben, dann sollten Sie der EDV größte Aufmerksamkeit widmen.
Bereits im privaten Sektor, der Domäne der Homecomputer, kann einiges getan werden.
Irgendwann ändert sicht die Struktur des Chemie-Studiums. Es wird wohl so aussehen, daß ein Grundwissen geschaffen wird, im Umfang vielleicht sogar etwas kleiner, als bisher. Mit höherer Semesterzahl wird es sich wie das Geäst eines Baumes, in verschiedene Spezialrichtungen verzweigen.

IN DER FORSCHUNG IST DER COMPUTER KEIN JOBKILLER

Der Druck, derartige Überlegungen anzustellen, kommt von der Computertechnik. Ein Programmierer ist kein Chemiker, und der Chemiker ist kein Informatiker. Bislang kann man sich mit Eigeninitiative, die bereits beim Studium beginnt, weiterhelfen. Aber die Anforderungen der EDV werden mit jedem neuen Rechner größer und sind nebenher nicht mehr lange zu bewältigen.
Der naturwissenschaftliche Informatiker ist noch nicht in Sicht und trotzdem kommt dem Computer eine immer größere Bedeutung zu. Von einem Jobkiller kann nicht die Rede sein. Software heißt das eigentliche Problem, mit dem man auch in der GSF zu kämpfen hat. Eine Chance für jeden Interessierten, mag er sich auch jetzt noch mit BASIC plagen.
Wer in der Computeranwendung seine Zukunft sieht, muß nicht unbedingt den Weg über das Studium einschlagen. Es besteht auch Bedarf an Laboranten jeglicher Art. Und auch hierbei ist, wie es so schön heißt, Erfahrung mit dem Computer kein Hindernis. Augenzwinkernd bedeutet dies wohl, daß Vorbildung keine finanziellen Folgen hat. Aber was macht das schon, wenn man im Berufsleben auch noch den Kontakt zu seinem Hobby pflegen kann? *GS*□

Mit zehn Mark sind Sie dabei: Drucken mit jedem Modell



Man sollte es eigentlich kaum glauben. Auf dem Markt werden Interfaces für 200 bis 300 DM angeboten, doch mit ein bißchen Material, das nicht einmal 10 DM kostet, und einem kleinen Programm können wir uns bereits ein Centronics-Interface für unseren C16/116/Plus4 basteln.

Ursprünglich wollten wir eigentlich die Centronics-Schnittstelle am Kassettenport realisieren. Wir dachten, daß der Kassettenport mit dem seriellen Port nichts zu tun hätte, weil im Betriebssystem des C16 eine Routine „Kassettenport einschalten" existiert.
Bei näherem Hinsehen jedoch merkten wir, daß hier gar keine Umschaltung auf den Kassettenport vorlag, sondern lediglich der Bildschirm ausgeschaltet und ein Timer gesetzt wurde. Durch Aufruf dieser Routine fand also keine elektrische Trennung von seriellem Port und Kassettenport statt. Bit Nummer eins in Adresse eins steuert also sowohl den Clock-Ausgang des seriellen Ports und die Schreib-Leitung des Kassettenports. Damit würde die serielle Ausgabe von Daten auf die Floppy 1541 also über unser geplantes Interface auch den Centronics-Drukker ansprechen, was wir nur allzugerne vermieden hätten. Da dieses sich leider nun sowieso nicht vermeiden läßt, wählten wir nun den seriellen Port für unser Interface aus, da wir hier nicht nur eine, sondern gleich drei Schreibleitungen zur Verfügung haben, was die Schaltung und vor allem die Programmlogik wesentlich vereinfacht.
Der Artikel über das Schieberegister zeigt uns bereits, wie wir mit nur drei Ausgabeleitungen acht Datenleitungen und eine Strobeleitung an den Drucker realisieren können. Für die Busy-Leitung hätten wir nun zwei Lösungen, nämlich doch noch den Kassettenport zu benützen oder aber eine unserer drei Leitungen am seriellen Port, mit denen wir nicht nur Daten ausgeben, sondern auch Daten lesen können, noch zusätzlich zur Abfrage des Busy-Signals zu verwenden.

ABFRAGEN DER BUSY-LEITUNG

Daß wir zur Erzeugung von acht parallelen Datenbits ein Schieberegister mit seriellem Eingang und parallelem Ausgang benötigen, dürfte wohl nicht all zu schwer zu erraten sein. Was noch ein Problem darstellt, ist, wie wir wohl das Busy-Signal auf eine Datenleitung bekommen, ohne daß durch dieses unsere auszugebenden Daten beeinflußt werden können. Der Umstand, daß das Busy-Signal bei der Datenausgabe immer auf Low gesetzt ist, und erst nach dem Zurücksetzen des Strobe-Signals den Wert High anzunehmen pflegt, kommt uns hierbei sehr gelegen. Mit einer Diode, die den Strom nur in einer Richtung durchläßt, könnten wir auf billige Weise den Einfluß des Strobe-Signals auf unsere Daten ausschalten. Die Diode allein nutzt uns aber noch nicht allzu viel. Sehen wir uns den Aufbau unseres seriellen Ports etwas näher an, so kommen wir sehr schnell darauf, daß noch ein weiterer elektronisches Bauteil vonnöten ist. Die Datenleitungen des seriellen Ports liegen über einen Pull-Up-Widerstand auf High, sofern durch entsprechen-

des Poken in Adresse eins die Ausgänge hochohmig sind. Durch Setzen der Bits null bis zwei auf den Wert eins wird dieser Zustand erreicht. Die Bits fünf bis sieben, mit Hilfe derer wir die anliegende Spannung lesen können, geben uns darüber Aufschluß. Es ist hierbei zu bemerken, daß Bit sieben mit Bit null korrespondiert, Bit sechs mit Bit eins und Bit fünf mit Bit zwei, wobei Bit null und sieben für die Daten-Leitung, Bit eins und sechs für die Clock-Leitung und Bit zwei und fünf für die ATN-Leitung zuständig sind. Die Low-Setzung einer Ausgabeleitung durch Setzen einer Eins in den entsprechenden

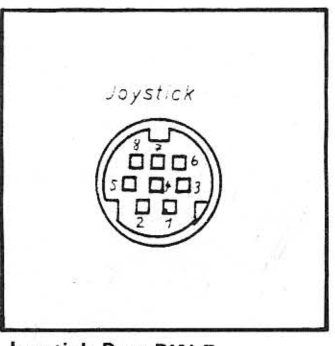
EXTERNE 5-VOLT-SPANNUNGS-VERSORGUNG

Bit oder das Anlegen von Low durch eine externe Verbindung sorgt dafür, daß die Spannung, die an den einem Ende des ein Kilo-Ohm großen Widerstandes noch fünf Volt betragen hatte, am anderen Ende auf null Volt heruntergezogen werden kann. Zum Lesen eines Signales ist somit der Ausgang hochohmig zu machen, und abzufragen, ob ein externes Signal eine Änderung des High-Zustandes bewirke oder nicht. Da aber das Busy-Signal, durch eine Diode gesperrt, nur die Werte hochohmig oder High annehmen kann, und somit immer der Wert eins gelesen würde, ist mit den erwähnten Bausteinen das Lesen des Busy-Signales leider noch nicht möglich. Wenn das Busy-Signal sich allerdings genau umgekehrt verhielte, nämlich die Werte Low oder hochohmig annehmen könnte, so bräuchten wir nur die Diode umzupolen, mit Sperrichtung zum Drukker und das Problem wäre gelöst. Würde der Drukker nach Zurücksetzen des Strobe-Signales das Busy auf Low setzen, so würde der Strom nun vom

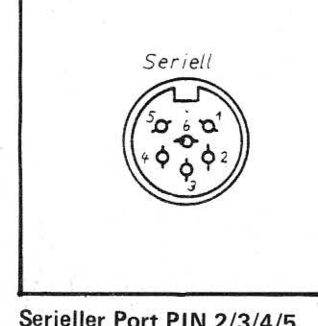
Anschlußtabelle

Port	74LS164	Diode	Inverter	Drucker
4	1			
4	2			
	3			2
	4			3
	5			4
	6	•		5
2	7		7	16
5	8			
	9			(18)
	10			6
	11			7
	12			8
	13			9
	14		14	(18)
4		>I		
		>I	2	
			1	1
3				11

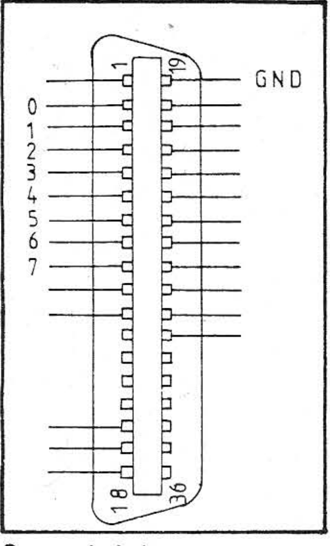
Pinbelegung des seriellen Ports des Rechners: Über den 74LS164 TTL-Baustein, Diode, Inverter zu einem beliebigen Centronics-Drucker.



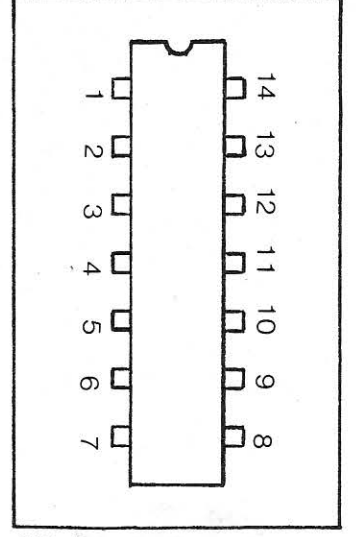
Joystick Port PIN 5



Serieller Port PIN 2/3/4/5



Centronicsbelegung



TTL-Baustein mit 14 Pins

<u>Was Sie hierzu brauchen:</u> Elektronische Bauteile für ca. 2 DM, Stecker und Platine für ca. 5 DM, etwas Kabel und Geduld beim Löten

Rechner zum Drucker fließen, somit die Spannung an unserem Eingang abfallen, und wir könnten diesen Zustand durch Abfragen des entsprechenden Bits wahrnehmen. Ginge das Busy-Signal im Anschluß auf High zurück, so bemerkten wir auch dieses wieder durch unsere Abfrage. Da von High zu High sowieso kein Strom fließt, und der Stromfluß von High nach Low durch die Diode unterbunden ist, hätte im Anschluß das Strobe-Signal keinen Einfluß mehr auf an das Schieberegister angelegte Spannungen. Was wir also nur zu tun hätten, wäre das Umdrehen oder Invertieren der Busy-Leitung. Hierzu aber gibt es auch ein elektronisches Bauteil, nämlich den Inverter, als letztes, uns noch fehlendes, Glied zur Realisierung unserer Schaltung.

ELEKTRONISCHE BAUTEILE

Die elektronischen Bauteile bekommen wir im Elektronik-Fachgeschäft. Falls an Ihrem Wohnort keines vorhanden sein sollte, so erfahren Sie sicherlich in einer Elektronik-Fachzeitschrift die Adresse eines Elektronik-Versandgeschäftes. Was Sie beim Kauf oder bei der Bestellung der Bausteine wissen müssen, ist deren Bezeichnung. Als serielles Schieberegister mit serieller Eingabe und paralleler Ausgabe kommt in Betracht der TTL-Baustein 74LS164. Eine schnelle niederohmige Diode mit geringem Spannungsabfall ist die Germaniumdiode AA112. Zum Invertieren des Busy-Signales können wir einen Inverter, zum Beispiel den TTL-Baustein 74LS04 oder auch ein NAND-Gatter zum Beispiel den TTL-Baustein 74LS03 verwenden. Inverter wie auch NAND-Gatter war in unserem Elektronik-Geschäft für

0,50 DM zu bekommen, die Diode für 0,49 DM und das Schieberegister für 1,10 DM. Für ca. 2,10 DM haben wir also unsere Elektronik beieinander. Eine gelochte Platine ohne Leiterbahnen ist für 0,60 DM zu haben, der Centronics-Stecker

CENTRONICS C16

```
10 rem centronics==============c16 <hl>
20 rem (p) commodore welt team <ho>
30 rem ========================== <ng>
40 rem (c) by alfons mittelmeyer <cg>
50 rem <pd>
60 rem <ah>
70 rem c16/116/plus4 <ja>
80 rem + centronicsinterface <gc>
90 rem ========================== <jg>
100 fori=1015to1067 <df>
110 reada:pokei,a:next <ne>
120 fori=1630to1729 <gl>
130 reada:pokei,a:next <gp>
140 fori=818to858 <kp>
150 reada:pokei,a:next:sys818:new <ki>
160 data 201,013,208,007,072,169 <el>
170 data 010,032,002,004,104,072 <le>
180 data 133,245,165,001,162,008 <lk>
190 data 009,003,006,245,144,002 <gn>
200 data 041,253,133,001,041,254 <aa>
210 data 133,001,202,208,239,009 <kp>
220 data 004,133,001,041,249,133 <ge>
230 data 001,036,001,080,252,024 <bk>
240 data 174,061,001,104,096 <im>
250 data 072,165,153,201,003,208 <ih>
260 data 089,165,173,201,116,208 <ok>
270 data 083,104,142,061,001,201 <cf>
280 data 017,208,002,024,096,170 <jh>
290 data 041,127,201,065,048,012 <kk>
300 data 201,091,016,008,224,000 <ni>
310 data 048,005,073,032,208,001 <fa>
320 data 138,174,061,001,076,247 <gg>
330 data 003,166,174,224,004,208 <pf>
340 data 008,162,003,134,174,162 <nd>
350 data 116,134,173,076,178,006 <ha>
360 data 166,173,224,116,208,004 <dm>
370 data 162,000,134,173,032,012 <ha>
380 data 239,162,000,134,173,096 <oi>
390 data 032,083,239,076,173,006 <ne>
400 data 032,093,238,076,173,006 <mi>
410 data 104,076,075,236 <ik>
420 data 169,160,141,032,003,169 <kc>
430 data 006,141,033,003,169,143 <gj>
440 data 141,024,003,169,006,141 <lb>
450 data 025,003,169,184,141,026 <bp>
460 data 003,169,006,141,027,003 <fe>
470 data 169,094,141,036,003,169 <ml>
480 data 006,141,037,003,096 <gf>
490 rem ========================== <pm>
500 rem p r o g r a m m e n d e <cp>
510 rem ========================== <af>
```

für 2,95 DM und ein sechspoliger Diodenstecker wird wohl auch nur 1,25 bis 1,95 DM kosten. Wenn wir nun nur noch unsere Kabel richtig verlöten, so ist unser Interface schon fertig.

Die Zahlen in der Tabelle geben die Nummern der Anschlußpins wieder. Wo die einzelnen Pins sitzen, können wir den nachfolgenden Bildern entnehmen.

Hierbei ist zu beachten, daß wir die Anschluß-

C16-CENTRONICS DRUCKER-INTERFACE

buchse am Computer so wiedergegeben haben, wie sie sich dem Auge von außen darbietet, den Centronics-Stecker dagegen ist von hinten gesehen, wo wir unsere Leitungen anzulöten haben. Die Dateneingänge des Schieberegisters liegen auf Pin eins und zwei. Da diese Anschlüsse intern an einem AND-Gatter anliegen, ist die vom Rechner kommende Leitung mit beiden Pins zugleich zu verbinden. Wir haben die Clock-Leitung an Pin vier des seriellen Ports als Datenleitung auserwählt, die Data-Leitung desselben Ports an Pin fünf dagegen als Clock-Leitung für das Schieberegister, da andersherum Probleme auftraten. Den Clock-Anschluß des Schieberegisters ist auf Pin acht zu finden. Als Strobe-Signal für den Drucker, das am Centronics-Stecker an Pin elf anzulegen ist, wählten wir das ATN-Signal des seriellen Ports an Pin drei. Die parallelen Ausgänge des Schieberegisters, Pins drei bis sechs und Pins 10 bis 13 sind an die Dateneingänge des Centronics-Steckers Pins zwei bis neun zu legen. Das Busy-Signal des Druckers auf Pin eins wird zum Eingang des Inverters, ebenfalls Pin eins geführt. Bei Verwendung des NAND-Gatters ist das Busy-Signal auf beide Eingänge des Gatters, Pin eins und zwei zu legen. Der Ausgang, Pin zwei beim Inverter bzw. Pin drei beim NAND-Gatter, kommt an das sperrende Ende der Diode, welches durch einen schwarzen Ring kenntlich gemacht ist. Das andere Ende der Diode verbinden wir mit einem unserer Schieberegistereingänge oder mit der Clock-Leitung des seriellen Ports, was auf dasselbe herauskommt. Da nun alle Signalleitungen verbunden sind, ist nur noch die Stromversorgung des Interface zu gewährleisten. Die Masse des Druckers, Pin 16, und die Masse des Rechners, Pin zwei, sind auf ein gemeinsames Potential zu bringen und mit den Pins sieben unserer zwei TTL-Bausteine zu verbinden. Die Versorgung mit 5 Volt ist etwas problematisch, denn nicht jeder Drucker hat die gewünschte Span-

HAT IHR DRUCKER 5 VOLT AUF PIN 18?

nung auch wirklich an Pin 18 anliegen. Wer Lust und Laune hat, durch ein wenig Löten im Drucker, diesem abzuhelfen, kann dieses gerne tun. Anderenfalls gibt es auch noch eine zweite, leicht zu realisierende Möglichkeit. Da die Benützung zweier Joystickports sowieso nur in speziellen Spielprogrammen vorkommt, liegen zumindest einer davon, wenn nicht alle zwei völlig brach. Wir können also ohne weiteres dort unsere Spannung abgreifen. Wir benötigen dazu nur ein isoliertes Kupferkabel von der nötigen Stärke, daß wir das abisolierte Ende gut in Pin fünf eines unserer Joystickports stecken können. Wenn die Spannung also nicht vom Drucker zu bekommen ist, dann bitte Pin 14 der TTL-Bausteine nicht mit Pin 18 des Centronics-Steckers verbinden sondern mit Pin fünf einer Joystick-Buchse.

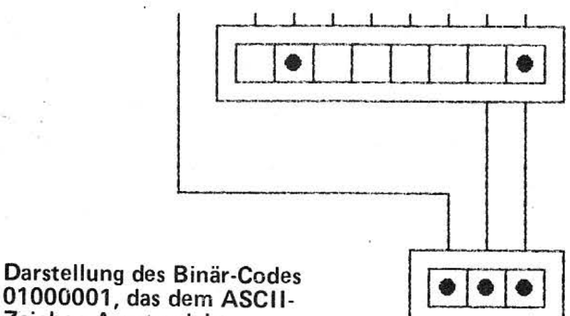
Wer eine Datasette oder eine Floppy 1551 benutzt, dürfte mit diesem Interface gut bedient sein. Für eine Floppy 1541 allerdings, die ebenfalls am seriellen Port betrieben wird, ist diese Lösung kaum vertretbar, denn Floppy und Drucker würden sich gegenseitig ins Gehege kommen. Mit ein paar Bausteinen mehr könnte auch hier Abhilfe geschaffen werden. Aber das soll nicht der jetzige Stoff sein.

SOFTWARE-TREIBER

Ausgaben über den seriellen Port laufen normalerweise völlig anders ab, als sie unser Interface benötigt. Es bleibt uns nichts anderes übrig, als das auszugebende Zeichen durch Verbiegen des Ausgabevektors BSOUT und eine geeignete Routine abzufangen und selbst die Ausgabe in geeigneter Weise zu veranlassen. Unser Programm führt eine ASCII-Umwandlung durch, damit Groß- und Kleinbuchstaben richtig ausgegeben werden, macht einen zusätzlichen Linefeed, damit auch diejenigen, die einen automatischen Linefeed am Drucker nicht einstellen können, keine Probleme damit haben und unterdrückt den ASCII-Code 17, womit Commodore-Drucker auf Groß-/Kleinschreibung umgestellt werden, andere Drucker aber zu unerwünschten Dingen, wie Einschalten irgendeines Grafik-Modus, veranlaßt würden. Wenn Sie unser Programm CENTRONICS-C16 gestartet haben, kann der Drucker auf Commodore-übliche Weise angesprochen werden. Commodore-spezifische Steuer- und Grafikzeichen allerdings sind von Ihnen durch die entsprechenden Drucker-spezifischen Zeichen zu ersetzen. Wir wünschen Ihnen viel Spaß mit Ihrem selbstgebauten Interface. *a.m.* □

Aus Zwei machAcht

Mit einem Schieberegister werden aus zwei Datenleitungen acht. Ein Demoprogramm zeigt nicht nur die Funktionsweise einer solchen Datenübertragung auf, sondern ermöglicht zu Textzwecken gar die Ausgabe von Daten.



Darstellung des Binär-Codes 01000001, das dem ASCII-Zeichen A entspricht.

Zehn Leitungen sind normalerweise nötig, um eine Centronics-Schnittstelle zu realisieren: nämlich acht Datenleitungen, die Strobe-Leitung, die dem Drucker das Bereitstehen von Daten signalisiert, und die Busy-Leitung, die uns die Bereitschaft zum Empfang weiterer Daten meldet. Sehen wir uns den seriellen Port unseres Rechners an, so entdekken wir nur drei Leitungen, die wir zum Senden und Empfangen benützen können.

AUS ZWEI MACH ACHT

Das sind wohl entschieden zu wenig, könnte man meinen. Doch glücklicherweise gibt es elektronische Bauteile, die dieses Manko beseitigen können. Was wir benötigen, ist ein Baustein, den wir seriell Bit für Bit mit Daten beschicken können und der diese schön parallel nebeneinander für den Drucker bereithält. Also lediglich ein Acht-Bit-Schieberegister mit seriellem Eingang und parallelem Ausgang, das es für eine Mark in jedem gut sortierten Elektronik-Fachgeschäft zu kaufen geben sollte.

Zur Ansteuerung eines solchen Schieberegisters sind nur zwei Leitungen erforderlich. Zum ersten eine Datenleitung, mit der wir entweder ein High- oder Low-Signal, je nach dem Zustand des zu übertragenden Bit, am Dateneingang des Registers anlegen können. Zum zweiten eine Taktleitung, durch die wir mit einem kurzen Signal, dem sogenannten Clocksignal, die Datenaufnahme auslösen. Nachdem sich nach acht Übertragungen die acht Bit des zu übertragenen Byte im Schieberegister befinden, ziehen wir kurz die dritte Leitung, die Strobe-Leitung, auf Low, und der Drucker hat seine Daten. Wie wir die Busy-Leitung abfragen, soll uns im Moment noch nicht interessieren. Eventuell können wir hierzu entweder die Daten- oder die Clock-Leitung hernehmen.

CENTRONICS-DEMOPROGRAMM

Da eine Demonstration oft mehr zu zeigen vermag, als viele Worte auszusagen versuchen, haben wir für Sie ein Demoprogramm verfaßt, welches wir jetzt starten wollen.
Unten am Bildschirm ist der Zustand unserer Leitungen durch einen gesetzten oder nichtgesetzten schwarzen Punkt symbolisiert. Die Strobe-Leitung links liegt normalerweise auf High. Die Clock- und Daten-Leitung liegen noch auf Low. Mit den beiden letztgenannten Leitungen steuern wir das Schieberegister, das noch gänzlich leer vor uns liegt.
Indem wir die Tasten [0] oder [1] drücken, legen wir unsere Daten an das Schieberegister an. Mit dem Druck der Leertaste lösen wir das Clock-Signal aus, das unser Datenbit in das Schieberegister bringt. Die nachfolgende Tabelle gibt uns Aufschluß über die Bit-Codes einiger ASCII-Zeichen.

	0010	0011	0100	0101
0000:		0	@	P
0001:	!	1	A	Q
0010:	"	2	B	R
0011:	#	3	C	S
0100:	$	4	D	T
0101:	%	5	E	U
0110:	&	6	F	V
0111:	'	7	G	W
1000:	(	8	H	X
1001:	)	9	I	Y
1010:	*	:	J	Z
1011:	+	;	K	[
1100:	,	<	L	Ł
1101:	–	=	M	J
1110:	.	>	N	^
1111:	/	?	O	

Über jeder Spalte befinden sich die vier ersten zu übertragenden Bit, am Anfang jeder Zeile die vier restlichen Bit. Nachdem das einem ASCII-Zeichen entsprechende Bitmuster im Schieberegister steht, löst ein Druck auf die Return-Taste das Strobe-Signal aus. Dadurch wird das Zeichen auf den Drukker, in unserem Demoprogramm hingegen in die erste Bildschirmzeile, ausgegeben.
Wenn Sie CW-EXTRA auf den Bildschirm zu bringen vermögen, so dürften Sie das Funktionsprinzip einer Centronics-Schnitt-

stelle am C16/116 schon voll erfaßt haben.

AUTOMATIK-MODUS

Eine Möglichkeit, die unser Demoprogramm noch bietet, haben wir Ihnen bis jetzt noch verschwiegen. Mit der ESC-Taste können Sie in den Automatik-Modus um- und von dort auch wieder zurückschalten. Im Automatik-Modus können wir hintereinander bis zu zehn Tasten drücken, denn soviel faßt der Tastaturpuffer. Nach der Eingabe setzt der uns bereits vertraute Ablauf programmgesteuert ein, und bald darauf ist der Buchstabe auf dem Bildschirm sichtbar.

Als wertvolle Hilfe beim Bau des Interface erwies sich die im Centronics-Demoprogramm vorgesehene Möglichkeit, die Daten auch wirklich am seriellen Port auszugeben. Hierzu sind nur die entsprechenden REMs, die die Ausgabe normalerweise verhindern, zu beseitigen.
Falls wir im Besitz eines Centronics-Schnittstellen-Testers sind, können wir an dessen leuchtenden Lämpchen ablesen, ob das Interface auch wirklich das tut, was auf dem Bildschirm zu sehen ist. Vollführt es das Erwartete, so steht nichts mehr im Wege, Daten an den Drucker zu senden.

a.m. □

CENTRONICS-DEMO

```
10 rem centronics-demo ========c16 <ip>
20 rem (p) commodore welt team <ho>
30 rem ========================== <ng>
40 rem by alfons mittelmeyer <nm>
50 rem <pd>
60 rem <ah>
70 rem basic v3.5 <nl>
80 rem c16/116/plus4 <ki>
90 rem ========================== <jg>
100 scnclr:gosub630 <jf>
110 char,7,4,"B "+lo$+"C"+u$+"C"+u
$+"C"+u$+"C"+u$+"C"+u$+"C"+u$+"C"+
u$+"C"+u$+"C"+ro$ <ka>
120 char,7,5,"B B"+lo$+"C"+o$+"C"+
o$+"C"+o$+"C"+o$+"C"+o$+"C"+o$+"C"
+o$+"C"+ro$+"B" <kn>
130 char,7,6,"B BB B B B B B B B B
B" <co>
140 char,7,7,"B B"+lu$+"C"+u$+"C"+
u$+"C"+u$+"C"+u$+"C"+u$+"C"+u$+"C"
+u$+"C"+ru$+"B" <oh>
150 char,7,8,"B "+lu$+"CCCCCCCCCC
CC"+o$+"C"+o$+"C"+ru$ <ci>
160 for i=9to12:char,7,i,"B":char,
23,i,"B B":next <oh>
170 char,7,13,lu$+"CCCCCCCCCCCC"+
ro$+" B B" <bm>
180 for i=14to15:char,21,i,"B B B"
:next <an>
190 char,19,16,lo$+"C"+u$+"C"+u$+"
C"+u$+"C"+ro$ <lm>
200 char,19,17,"B"+lo$+"C"+o$+"C"+
o$+"C"+ro$+"B" <ok>
210 char,19,18,"BBQBQB BB" <ke>
220 char,19,19,"B"+lu$+"C"+u$+"C"+
u$+"C"+ru$+"B" <hd>
230 char,19,20,lu$+"CCCCCC"+ru$ <gp>
240 a$=" ":fori=1to7:a$=a$+l$+" ":
next <hf>
250 cl=0:da=0:sr=-1 <oh>
260 getkeyx$ <bf>
270 ifx$="0"thenda=0 <oo>
280 ifx$="1"thenda=-1 <ec>
290 ifx$=" "thencl=-1 <bm>
300 ifx$=chr$(13)thensr=0 <dg>
310 ifx$=chr$(27)thengoto580 <ii>
320 gosub330:goto260 <me>
330 ifsrthen430 <hc>
340 char,21,18," ":fori=1to500:nex
t:char,21,18,"Q" <fd>
350 b=0:fori=1to15step2:b=b+b <dk>
360 ifmid$(a$,i,1)="Q"thenb=b+1 <cf>
370 next:sr=-1:fork=1to500:next <on>
380 char,x,y,chr$(b):x=x+1 <nm>
390 rem -------------------------- <fj>
400 rem  strobe <pb>
410 rem -------------------------- <oa>
420 rem  poke1,peek(1)or4:poke1,pe
ek(1)and251 <aj>
430 ifdathenda$="Q":elseda$=" " <ei>
440 rem -------------------------- <hf>
450 rem  data <go>
460 rem -------------------------- <cj>
470 rem  ifdathenpoke1,peek(1)and2
53:elsepoke1,peek(1)or2 <nl>
480 ifnotclthen550 <no>
490 char,23,18," ":fori=1to500:nex
t:char,23,18,"Q":cl=0 <mk>
500 a$=right$(a$,13)+l$+da$ <gl>
510 rem -------------------------- <mb>
520 rem  clock <bf>
530 rem -------------------------- <ah>
540 rem poke1,peek(1)or1:poke1,pee
k(1)and254 <jp>
550 char,25,18,da$ <kp>
560 char,11,6,a$ <io>
570 return <ag>
580 getkeyz$:z=asc(z$):ifz=27then2
60 <pg>
590 forj=1to8:z=z+z:fork=1to500:ne
xt <je>
600 da=0:ifz>=256thenda=-1:z=z-256 <ab>
610 gosub330:cl=-1:fork=1to500:nex
t:gosub330:next:fork=1to500:next <eb>
620 sr=o:gosub330:goto580 <hp>
630 l$=chr$(29):lo$=chr$(176) <eo>
640 ro$=chr$(174):lu$=chr$(173) <pb>
650 ru$=chr$(189):o$=chr$(178) <mb>
660 u$=chr$(177):return <ok>
670 rem ========================== <fh>
680 rem  p r o g r a m m e n d e <ji>
690 rem ========================== <cn>
```



Kleine Show ganz groß

Vom 3. bis 7. Februar wurde das Dortmunder Ausstellungsgelände Westfalenhallen zum Treffpunkt für Computer-Freaks und Hobby-Elektroniker. Die Hobby-tronic & Computerschau '88 präsentierte als kombinierte Ausstellung ein elektronisches Riesenangebot für jedermann.

Hobby-tronic & Computerschau in Dortmund

Etwa 75 000 Besucher fanden in fünf Messetagen den Weg nach Dortmund zur Hobby-tronic & Computerschau. Nachdem im vorigen Jahr die Messeleitung, durch den großen Andrang bedingt, die Messe mehrmals schließen mußte, wurden in diesem Jahr die Gänge verbreitert und die Ausstellungsflächen vergrößert. Auch 1988 präsentierte sich die Ausstellung mit der im Vorjahr erfolgreichen Konzeption: In Halle 5 befand sich die Hobby-tronic, elfte Ausstellung für Funk- und Hobby-Elektronic, in Halle 6 war die Computer-Schau als vierte Ausstellung für Computer, Software und Zubehör aufgebaut. Diese Gliederung soll eine gute Übersicht und ein leichtes Auffinden der Produktgruppen ermöglichen.

An der Zwillingsmesse beteiligten sich 135 Aussteller, die 28 Firmen vertraten. Hinzu kamen mehrere Freizeit-Verbände des Computer- und Elektronik-Sektors, die über ihr Hobby firmen- und systemneutral unterrichteten.

75000 MESSEBESUCHER AUF DER HOBBY-TRONIC

Die Computer-Schau '88 bot Systeme aller gängigen Hersteller. Das Preis-Spektrum reichte von etwa 100 bis zu 25 000 Mark für aufwendig ausgestattete Systeme zur professionellen Anwendung. Computer-Besitzer fanden Programme von der Adressenverwaltung über Auftragbearbeitung und Textverarbeitung bis hin zur kompletten Finanzbuchhaltung vor, das alles im Preisbereich zwischen 49 und 199 Mark. Aber auch diejenigen, die nicht über einen so großen Etat verfügen, kamen auf ihre Kosten. Vor allem die Bastler konnten so manches Schnäpchen machen: defekte Computer zu Schleuderpreisen, 5 1/4-Zoll-Floppy-Laufwerke ab 25 Mark, dazu das umfangreiche Angebot an Zubehör.

Zwar wurde eine Vielzahl von Produkten vorgestellt und zum Verkauf angeboten, ein absoluter Messerenner jedoch fehlte. So war es auch nicht verwunderlich, daß hauptsächlich die Stände mit Software-Angeboten stark umlagert waren. Die Preispalette reichte von 5 Mark bis unendlich. Jeder, der etwas bestimmtes suchte, durfte fündig geworden sein.

Gut besucht waren die Stände der verschiedenen Computerclubs, die sich mit teilweise sehr hochwertigen und ausgefallenen Produkten präsentierten und ihre Clubarbeit vorstellten. Neulinge unter den Computerfans fanden in den Fachabteilungen der vertretenen Kaufhäuser die Computertypen gesondert vorgestellt und erläutert.

Besonders Commodore-Fans kamen auf ihre Kosten: Fast alle Zubehörfirmen, die für diese Computer produzieren, waren anwesend. Ein gebrauchter 6522 für 3 Mark, eine Epromplatine ab 5 Mark, eine 320-KByte-Erweiterung: Es war alles vorhanden, was das User-Herz begehrte. Den C16/116- und Plus4-Fans wurde das Centronics-Kabel (COMMODORE WELT 12/87) für den Plus4 vorgestellt, auch andere Soft- und Hardware für diese Computertypen gab's in Hülle und Fülle.

Durch die große Nachfrage waren einige Produkte (RAMs 41256, Amiga- oder Atari-Modulator), teilweise bereits nach dem dritten Tag ausverkauft.

ATARI & COMMODORE NICHT VERTRETEN

Auch Buch- und Zeitungsverlage ließen es sich nicht nehmen, ihre Erzeugnisse vorzustellen. Der Verbraucher konnte sich vor Ort über die Fülle von Fachliteratur informieren. In Halle 6 befand sich auch die neutrale „Computer-Beratung". Hinzu kam die „Computer-Straße", auf der sieben Computerclubs mit Tips, Hinweisen und Software ohne kommerzielles Interesse vertreten waren.

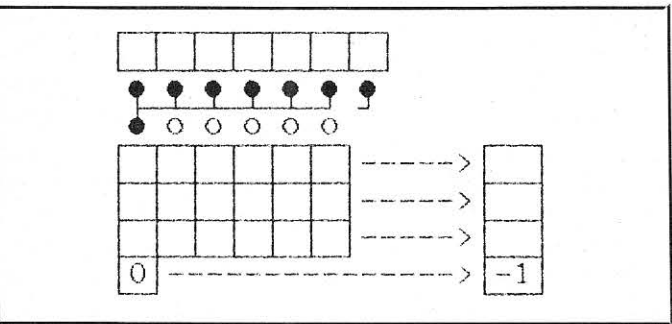
Interessant für alle, die den Einstieg ins Programmieren finden wollten, war das Kursprogramm zur Computer-Schau. Dreimal täglich wurden Kurse angeboten, die Programmier-Grundkenntnisse vermitteln sollten.

Allerdings waren die großen Homecomputer-Hersteller wie Commodore oder Atari nicht durch eigene Stände vertreten Wegen der CeBIT in Hannover haben sie wohl auf die Vorstellung der neuesten Produkte in Dortmund verzichtet.

bw□

Lernen am Programm

Programmlistings, gut kommentiert, können eine Fundgrube für den Anfänger sein. Oft sieht er sich aber mit einer Menge Listings konfrontiert, die so gemacht sind, als habe es der Autor darauf angelegt, jedes geistige Durchdringen zu verhindern. Am Beispiel einer Lottoauswertung, die wir haargenau erklären, zeigen wir, wie ein Programmierproblem gelöst wird.



Der Computer kann uns besonders bei stumpfsinnigen Tätigkeiten sehr viel Arbeit abnehmen, vor allem bei Auswertungen und Sortiervergängen. Manche Aufgabenstellungen scheinen zwar anfänglich etwas verzwickt, aber nach einer eingehenden Analyse läßt sich meist die ganze Sache vereinfachen und ein klares Konzept für die Problemlösung erstellen.

PROBLEMSTELLUNG

"Ich habe folgendes Problem: Ich benötige ein Programm, bei dem ich 100 Zeilen mit jeweils sechs Zahlen eingeben kann. Wenn ich nun sechs verschiedene Zahlen inklusive Zusatzzahl eingebe, soll mir der Computer alle Dreier, Vierer, Fünfer, Fünfer mit Zusatzzahl und Sechser anzeigen, die sich aus den 100 mal sechs Zahlen ergeben.

Eine Aufteilung der Aufgabenstellung in kleinere Teilschritte erhalten wir bereits, wenn wir die Grundbegriffe der EDV, also Dateneingabe, Datenverarbeitung und Datenausgabe, heranziehen.

DATENEINGABE

Sechs Zahlen und eine Zusatzzahl mit Hilfe einer INPUT-Anweisung zu erfassen, dürfte kein allzu großes Problem sein. 600 Zahlen, wobei ja auch Tippfehler auftreten könnten, bereiten uns da schon mehr Schwierigkeiten. Diese können wir aber sehr einfach dadurch umgehen, daß wir von der DATA-Anweisung Gebrauch machen und unsere Daten in DATA-Zeilen ablegen. Uns allerdings auf genau 100 Zeilen mit der entsprechenden Anzahl von Tips festzulegen, würde die Verwendbarkeit des Programmes nur unnötig einschränken. Wir sollten die Anzahl der Tips völlig offenlassen. Um unserem Programm zu sagen, mit wie vielen Daten es zu tun hat, ist noch eine Endmarkierung unterzubringen. Da die Lottozahlen nur Werte von eins bis 49 umfassen, bietet sich die Null als Endemarke an. Wir schreiben also in jede DATA-Zeile sechs Zahlen und in eine weitere die Zahl Null. Damit ist der Eingabeteil bereits gelöst.

DATENAUSGABE

Die Datenausgabe vermag mit den eingegebenen Daten nicht allzuviel anzufangen. Es bedarf vorher noch der Datenverarbeitung, die in unserem Falle eine Auswertung der Lottotips darstellt. Hätten wir neben den Eingabedaten zusätzlich noch eine entsprechende Liste, die uns Aufschluß über die Anzahl der Richtigen zu jedem Tip geben würde, so könnten wir nun bereits an die Ausgabe denken. Hierzu bräuchten wir nur diese Liste zum Beispiel nach drei oder sechs Richtigen zu durchsuchen und Nummern – eventuell haben wir unsere Spiele durchnumeriert – und Zahlen der dazugehörigen Tips auszugeben. Durch Hochzählen einer Variablen könnten wir anschließend auch noch die Anzahl unserer richtigen Tips auf den Bildschirm bekommen.
Wenn wir lesen, wir hätten zwanzig Vierer, mag uns dies zu einem Freudenschrei Anlaß geben. Da zumindest bei den Dreiern die Auflistung von mehr Tips, als der Bildschirm wiederzugeben vermag, möglich sein kann, empfiehlt es sich, eine Stoppfunktion durch Druck der Leertaste zu implementieren. Was uns jetzt bei unseren Überlegungen noch fehlt, ist die Auswertung, die uns unsere Liste mit der Zahl der Richtigen erstellt.

DATENVERARBEITUNG

Das Prinzip ist einfach zu verstehen: Wir brauchen nur nachzusehen, wieviel Richtige unser Tip enthält und das Ergebnis in die Liste einzutragen. Wenn wir dieses Verfahren sukzessiv durchgeführt haben, bis alle Tips eingetragen sind, ist die Verarbeitung beendet. Jedoch wäre die Anweisung, die Anzahl unserer Richtigen festzustellen, für den Computer sehr unklar. Wir müssen sis ihm erst noch plausibel machen.

DATENVERGLEICH

Um festzustellen, ob ein Element einer Liste in einer anderen Liste enthalten ist, bedarf es nicht nur eines einzigen Blickes, sonder einer Anzahl von Vergleichen. Das Element der einen Liste muß mit jedem Element der anderen verglichen werden. Bei Feststellung der Identität kann entweder ein Zähler erhöht oder ein Flag gesetzt werden. Wir vergleichen ein DATA unseres Tips mit jedem der sechs ausgelosten und durch unsere INPUT-Anweisung erfaßten Werte der Richtigen. Bei Identität der Werte dürfen wir einen Zähler erhöhen. Bei Identität mit der siebten Zahl, der Zusatzzahl, setzen wir ein Flag.
Wenn wir auf diese Weise jedes unserer DATAS mit den richtigen Lottozahlen verglichen haben, dürfen wir uns anschicken, die Anzahl unserer Richtigen in unserer zu erstellenden Liste zu erfassen. Bei fünf Richtigen kommt noch eine Sonderüberprüfung hinzu, nämlich, ob eventuell das Zusatzzahlflag gesetzt ist. Wenn ja, so müssen wir diesen Fall kenntlich machen und dürfen nicht lediglich fünf Richtige eintragen. Es bietet sich hierzu als Kennung die Zahl sieben an. Sind alle Tips verarbeitet, so sollte unsere Auswertungsliste auch eine Endemarke erhalten. Hierzu haben wir eine Eins mit negativem Vorzeichen gewählt, da eine weitere negative Zahl in unserer Liste nicht vorkommen kann.

Zeile 130:
Eingabeaufforderung.

Zeile 140:
Die sechs Lottozahlen werden mit einer indizierten Variablen A(K), mit K= 1 bis 6, erfaßt. Dies hat den Vorteil, daß später bei der Auswertung (vgl. Zeile 200) nicht jedes A(K) einzeln abgefragt werden muß, sondern mit Hilfe einer Laufschleife bearbeitet werden kann. Da der Zusatzzahl eine besondere Behandlung widerfährt, kann sie mit der Variablen Z erfaßt werden. Beim Programmablauf sind, durch die INPUT-Anweisung bedingt, sieben Zahlen, durch Komma voneinander getrennt, einzugeben, wobei die letzte die Zusatzzahl sein muß.

Zeile 180:
Die Anzahl der Richtigen soll in einer Liste erfaßt werden. Wir verwenden dafür das Feld W%(I) und demensionieren es auf 101, was der maximalen Anzahl unserer Lottotips plus unserer Endmarkierung entspricht. Um keine Speicherstelle zu verschenken, lassen wir den Feldindex I nicht mit Eins, sondern mit Null beginnen. Die Vorbereitungen sind getroffen, und wir können nund die erste getippte Zahl des ersten Spieles mit einer READ-Anweisung einlesen. Wir wählen hierzu die Variable X.

Zeile 190:
Diese Zeile soll nicht nur für das erste DATA überhaupt, sondern für die erste Zahl in jeder DATA-Zeile gelten. Wir überprüfen hier, ob wir bereits das Ende unserer DATAS erreicht haben. Haben wir eine Null vor uns, so ist dies der Fall und wir vermerken nun auch in unserer Auswertungsliste W%(I), durch eine negative Eins, deren Ende. Da der Auswertungsteil damit abgeschlossen ist, springen wir weiter zum Ausgabeteil.

Zeile 200:
Hier geht es um die Auswertung eines Lottospieles.
Wir setzen zu Beginn den Zähler N für die Anzahl unserer Richtigen auf Null, ebenso das Zusatzzahlflag ZF. Unsere sechs Lotto-DATAS müssen nun mit den richtigen Zahlen verglichen werden. Wir veranlassen dies mit einer Schleife und der Laufvariablen J. Jede unserer Lottozahlen muß mit den sechs Richtigen verglichen werden. Eine weitere Schleife mit der Laufvariablen K sorgt hierfür. Nun macht sich die Indizierung der in Zeile 140 erfaßten Werte bezahlt. Bei Übereinstimmung der zu überprüfenden Zahl X mit einer der sechs Richtigen A(K) wird der Zähler N erhöht.

Zeile 210:
Die NEXT-Anweisung, die für den Vergleich unserer Zahl X mit der nächsten Richtigen A(X) sorgt, darf wegen der IF-Anweisung in der vorhergehenden Zeile erst jetzt erfolgen. Nachdem nun unsere Zahl mit den sechs Richtigen verglichen ist, überprüfen wir, ob sie vielleicht mit der Zusatzzahl übereinstimmt und setzen in diesem Fall das Zusatzzahlflag ZF auf Eins minus. Dieser Wert bietet sich an, weil er dem logischen Wert TRUE entspricht. In einer Abfrage (vgl. Zeile 220) genügt daher statt IFZF =−1 bereits IFZF.

Zeile 220:
Unsere Zahl X ist nun mit den richtigen Lottozahlen verglichen worden. Wir müssen nun auf dieselbe Weise unsere fünf weiteren Zahlen vergleichen. Unsere nächste Zahl X erhalten wir mit einer READ-Anweisung. NEXTJ sorgt für die Abarbeitung der neuen Zahl (vgl. Zeile 200), sofern unser Lottotip noch nicht restlos abgearbeitet ist. Anderenfalls steht uns mit X bereits die erste Zahl der nächsten DATA-Zeile zur Verfügung. Die Anzahl der Richtigen steht nun im Zähler N. Bei fünf Richtigen überprüfen wir durch zwei geschachtelte IF-Anweisungen, ob unser Zusatzzahlflag ZF gesetzt wurde und wir also auch die Zusatzzahl richtig haben. Ist dies der Fall – es wäre nur allzu schön –, so setzen wir zur Markierung dieses Sachverhaltes unseren Zähler N auf den Wert sieben.

Zeile 230:
Das ausgewertete Ergebnis N dürfen wir nun in unsere Liste W%(I) eintragen. Nachdem wir den Feldindex I erhöht haben, geht es weiter mit dem nächsten Lottotip durch den Sprung nach Zeile 190. Die neue Zahl X steht uns hierzu, wie wir bereits im Kommentar zur vorangegangenen Zeile bemerkten, schon zur Verfügung.

Zeile 270:
Da uns natürlich besonders interessiert, ob wir einen Glückstreffer haben, beginnen wir nicht mit drei, sondern mit sechs Richtigen und schreiben dies, mit einer Leerzeile Abstand zur Eingabe, auf den Bildschirm. Nachdem wir der Variablen K, zum Zeichen dafür, daß wir nach sechs Richtigen suchen, den entsprechenden Wert übergeben haben, soll eine Unterroutine ab Zeile 350 den Rest erledigen. Unterroutinen empfehlen sich überall dort, wo ein und derselbe Algorithmus mehrfach Verwendung findet, wie dies ja auch in Zeile 280 der Fall ist.

Zeile 280:
Analog zu Zeile 270. Da die Unterroutine ab Zeile 360 selbst für eine Leerzeile sorgt, fehlt die zusätzliche PRINT-Anweisung der Zeile 270.

Zeile 290:
Nach Abarbeitung der Fünfer mit Zusatzzahl liegt keine Sonderregelung mehr vor, und so können die restlichen Treffer mit einer Schleife unter Zuhilfenahme der Laufvariablen K, die nun abwärts von fünf bis drei läuft, weiter bearbeitet werden.

Zeile 300:
Mit der RIGHTS-Anweisung, in Verbindung mit STRS(K) und der LEN-Funktion, sorgen wir dafür, daß die Bild-

LOTTO-AUSWERTUNG

```
10 rem lottoauswertung=========cbm <he>
20 rem (p) commodore-welt-team     <ie>
30 rem ==========================  <ng>
40 rem (c) by alfons mittelmeyer   <cg>
50 rem                             <pd>
60 rem                             <ah>
70 rem basic v2.0                  <nc>
80 rem c16/116/plus4/c64/c128 u.a. <bn>
90 rem ==========================  <jg>
100 rem -------------------------- <op>
110 rem  eingabe                   <mj>
120 rem -------------------------- <po>
130 print"eingabe von 7 zahlen (6
+ zusatzzahl)"                     <fh>
140 inputa(1),a(2),a(3),a(4),a(5),
a(6),z                             <gk>
150 rem -------------------------- <jj>
160 rem  ermittlung der richtigen  <ld>
170 rem -------------------------- <ef>
180 dimw%(101):i=0:readx           <oj>
190 ifx=0thenw%(i)=-1:goto270      <if>
```

schirmausgabe der restlichen Treffer an den linken Rand zu liegen kommt. Da K bereits Laufvariable ist, braucht dieser Parameter nicht zusätzlich übergeben zu werden, sondern es kann sogleich der Aufruf der Unterroutine erfolgen.

Zeile 310:
NEXTK sorgt dafür, daß die Auswertungsergebnisse für die nächst geringere Trefferzahl ausgegeben werden (vgl. Zeile 290). Nach der Ausgabe der Dreier sind wir schließlich fertig, so daß die END-Anweisung das Programm beenden darf.

Zeile 350:
Da wir nicht nur sehen wollen, wie viele Treffer wir haben, sondern diese vielleicht zur Kontrolle mit unseren Lottoscheinen vergleichen möchten, sollten auch die Daten unserer gewonnenen Spiele nochmals ausgegeben werden. Mit der RESTORE-Anweisung setzen wir den DATA-Zeiger auf den Anfang der DATAS. Den Feldindex I setzen wir auf Null, ebenso einen Zähler N, der uns Aufschluß darüber geben soll, wie viele Dreier, Vierer oder sonstige Treffer wir erreicht haben.

Zeile 360:
Das Flag S soll uns sagen, ob die in der Liste W%(I) erfaßte Zahl der Richtigen unseres Tips mit der Trefferzahl K, nach der wir suchen, übereinstimmt. ist dies der Fall, so erhöhen wir den Trefferzähler N und geben die Nummer des Spiels aus, die der um eins erhöhte Wert des Feldindexes ist. Wir haben diesen ja nicht bei eins, sondern bei Null beginnen lassen.

Zeile 370:

Mit einer Schleife von J=1 bis J=6 lesen wir die DATS unseres Tips ein, unabhängig davon, ob diese auszugeben sind oder nicht. Denn wenn wir ein Spiel finden, das unsere Bedingungen erfüllt, muß der DATA-Zeiger auch auf die richtigen Daten weisen. Durch IFS sorgen wir dafür, daß die Daten nur auf Wunsch ausgegegeben werden. Dem Ausdruck lassen wir zur besseren Trennung der einzelnen Zahlen noch ein Komma folgen.

Zeile 380:
NEXT sorgt für das Einlesen und den eventuellen Ausdruck der restlichen DATAS unseres Tips (vgl. vorige Zeile). Im Falle des Ausdrucks löschen wir mit PRINTCHRS (20), also dem ASCII-Wert der DEL-Taste, das letzte, unnötige, Komma und beginnen gleichzeitig eine neue Zeile.

Zeile 390:
Damit unsere Bildschirmausgabe nicht einfach nach oben wegSCROLLen kann und uns nicht genug Zeit zum Lesen läßt, sorgen wir für eine Stoppmöglichkeit. Wir fragen ab, ob die Leertaste gedrückt wurde. Ist dies nicht der Fall, so geht es weiter im Programm mit Zeile 410. Sonst folgt Zeile 400.

Zeile 400:
Solange die Leertaste nicht ein zweites Mal gedrückt ist, verharrt unser Programm in der durch Zeile 400 gegebenen Warteschleife.

Zeile 410:
Ob die DATAS unseres Tips ausgegeben sind oder nicht: Es geht nun an die Untersuchung des nächsten Tips. Hierzu erhöhen wir den Feldindex I. Weiter geht es dann wieder mit Zeile 360, sofern wird das Listenende, welches mit Eins minus gekennzeichnet ist, noch nicht erreicht haben.

Zeile 420:
Am Ende der Lise angelangt, geben wir die Anzahl unserer Treffer aus. Nach einer abschließenden Leerzeile kehren wir mit RETURN zum Hauptprogramm zurück, wo uns vielleicht noch weitere Ausgaben, die ebenfalls von der Unterroutine Gebrauch machen, erwarten.

Zeile 460:
Ab hier sind die Daten unserer Lottotips einzutragen. Zu Testzwecken ist dies in den Zeilen 460 - 490 schon einmal geschehen.

Zeile 530:
Das Ende der Daten ist mit einer Null zu kennzeichnen. Die Zeile 530 soll uns daran erinnern, dies auch am Schluß der Eintragung unserer wirklichen Tips zu tun.

```
200 n=0:zf=0:forj=1to6:fork=1to6:i
fx=a(k)thenn=n+1                 <jg>
210 next:ifx=zthenzf=-1          <fl>
220 readx:nextj:ifn=5thenifzfthenn
=7                               <pl>
230 w%(i)=n:i=i+1:goto190        <lh>
240 rem ------------------------- <ch>
250 rem  ergebnisse              <pp>
260 rem ------------------------- <ob>
270 print:print"6-er":k=6:gosub350 <eo>
280 print"5-er mit zusatzzahl":k=7
:gosub350                        <gg>
290 fork=5to3step-1              <hb>
300 printright$(str$(k),len(str$(k
))-1)"-er":gosub350              <kk>
310 nextk:end                    <kf>
320 rem ------------------------- <kp>
330 rem  unterroutine            <lh>
340 rem ------------------------- <ip>
350 restore:i=0:n=0              <ib>
360 s=w%(i)=k:ifsthenn=n+1:print"s
piel"str$(i+1)":";               <pb>
370 forj=1to6:readx:ifsthenprintst
r$(x)",";                        <ng>
380 next:ifsthenprintchr$(20)    <pb>
390 getx$:ifx$<>" "then410       <mm>
400 getx$:ifx$<>" "then400       <mn>
410 i=i+1:ifw%(i)<>-1then360     <ck>
420 print"anzahl:"n:print:return <ch>
430 rem ------------------------- <be>
440 rem  lottodaten              <am>
450 rem ------------------------- <ni>
460 data 1,3,5,7,9,11            <ad>
470 data 2,4,6,8,10,12           <gk>
480 data 1,2,3,4,5,6             <pb>
490 data 8,9,10,11,12,13         <pl>
500 rem ------------------------- <on>
510 rem  endemarkierung          <ha>
520 rem ------------------------- <pg>
530 data 0                       <ih>
540 rem ========================= <mm>
550 rem  p r o g r a m m e n d e <kk>
560 rem ========================= <hk>
```

Schnell, schneller am schnellsten

Meist gibt es zu einem Weg auch eine Abkürzung. Wir zeigen Ihnen, wie aus einem Programm durch zwei zusätzliche Zeilen ein Autostartprogramm wird. Sind Ihre BASIC-Programme zu langsam? Wir helfen Ihnen, sie schneller zu machen. Hilfestellung für Spieleprogrammierer: Darstellung von Figuren mit Strings. Programmieren mit Script/Plus?

In der letzten Ausgabe wurde ein Programm abgedruckt, mit dem aus einem normalen Programm ein Autostartprogramm gemacht werden kann. In der Zwischenzeit sind wir auf einen wesentlich einfacheren Weg gestoßen: Wie Sie sicher wis-

AUTOSTART MIT ZWEI ZEILEN

sen, kann die Funktionstaste F1 über zwei Speicherstellen aktiviert werden. In 1373 steht die Anzahl von Zeichen, die noch abgearbeitet werden sollen, und in 1374 die Startposition im Funktionstastenspeicher. Damit läßt sich ein BASIC-Programm auf einfache Weise zu einem Selbststarter machen.
Zuerst wird die Funktionstaste F1 mit dem RUN-Befehl belegt:

```
KEY1,CHR$(019)+"rU:"
<RETURN>
```

Der Doppelpunkt ist nötig, damit nachfolgende Zeichen keinen Syntax-Error auslösen. Jetzt kann noch ein beliebiger Text oder eine LORES-Grafik (mit dem eingebauten Grafik-Zeichensatz) auf den Bildschirm gezeichnet werden. Auch Farbe ist möglich, da der Farbspeicher ebenfalls mit abgespeichert wird. Jetzt kann alles zusammen mit den Funktionstasten und den beiden Speicherstellen zur Steuerung der Funktionstasten abgespeichert werden:

```
POKE44,5:POKE43,93:
POKE1373,4:POKE1374,
0:DSAVE"name"
<RETURN>
```

Nachdem das Programm abgespeichert ist, startet es sich automatisch.
Am besten wird der Rechner kurz ausgeschaltet, um das Ergebnis zu testen. Das unter "name" abgespeicherte Programm muß nun mit:

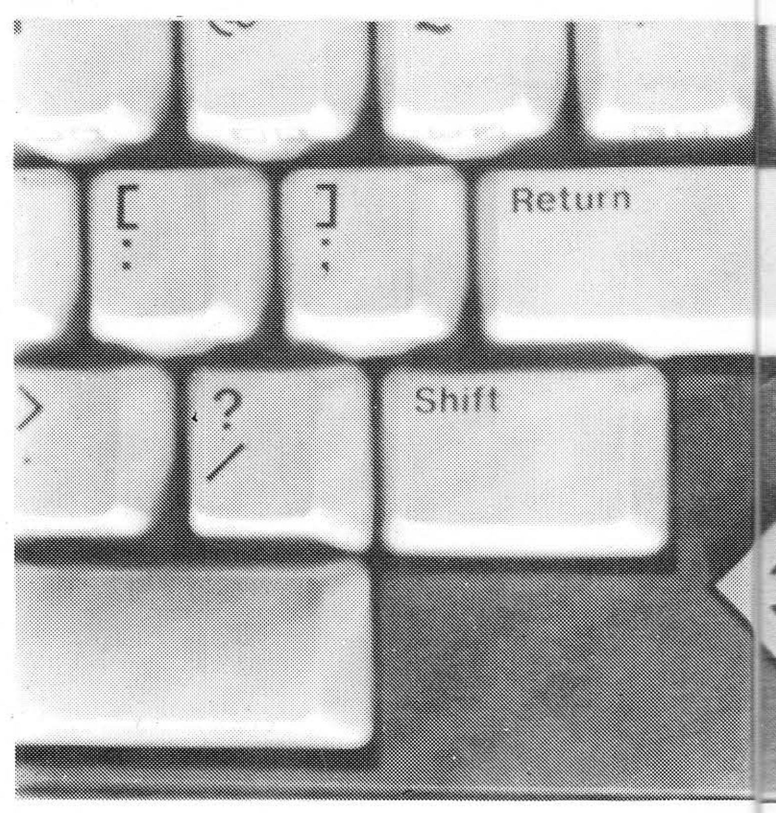
```
LOAD"name",8,1
<RETURN>
```

absolut geladen werden. Zuerst erscheint beim Laden der Text auf dem Bildschirm, und nach Beendigung des Ladevorgangs startet es sich von selbst. Einfacher geht es nicht mehr.

BASIC-PROGRAMME AUF TRAB GEBRACHT

Selbst wer das Handbuch auswendig kennt und alle Befehle vollständig beherrscht, ist noch lange kein perfekter Programmierer. Ein gutes BASIC-Programm wird vor allem von zwei Faktoren bestimmt:
a) übersichtliche Programmierung: Jeder andere Programmierer sollte das Programm verstehen können. Zwar ist dies bis zu einem gewissen Grad eine Geschmacksfrage, jedoch unserer Meinung nach eine wichtige Frage der Programmierung.
b) Geschwindigkeit des Programmablaufs: Sie wird ausschließlich durch den Interpreter beeinflußt und kann durch allgemein gültige Regeln optimiert werden.

Um den Ablauf eines BASIC-Programms zu beschleunigen, sind Kenntnisse über die Vorgehensweise des Interpreters unerläßlich. Das Programm steht verschlüsselt im Programmspeicher. Jedem Befehlswort und jedem



GESCHWINDIGKEIT IST KEINE HEXEREI

ASCII-Zeichen ist eine bestimmte Zahl zugeordnet, die der Interpreter in eine für den Computer verständliche Sprache übersetzen muß. Außerdem zeigt eine Null im Speicher das Ende einer BASIC-Zeile an. Dieser folgen vier Speicherstellen, in denen die Adresse der Folgezeile und die Zeilennummer stehen. Der Interpreter muß also Zeichen für Zeichen aus dem Speicher lesen und richtig umsetzen. Das braucht Zeit. Indem besonders langwierige Vorgänge vermieden werden, kann die Programmgeschwindigkeit drastisch beschleunigt werden.

Zeitmessung mit TI$

Die vom Interpreter benötigte Zeit kann leicht gemessen werden: Die interne Uhr wird mit TI$ ="000000" auf Null gestellt. Wenn die zu untersuchenden Befehle durchgeführt sind, wird die Zeit mit PRINT TI ausgegeben.

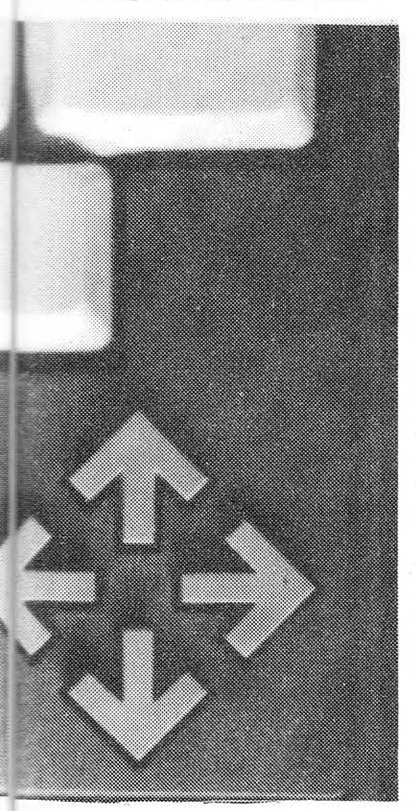
ZEIT, WER HAT DIE SCHON?

Allerdings läßt sich die Zeit, die für einen einzigen Befehl gebraucht

wird, nicht messen, sondern nur eine Kette von Befehlen gleichzeitig. Als Zeitmaß wollen wir im weiteren eine 1/60stel Sekunde als eine Zeiteinheit bezeichnen (PRINT TI gibt die Zeit so an).

In der Kürze liegt die Würze

Da alles, was im Speicher steht, auch abgearbeitet werden muß, sollten unnötige Leerzeichen und



REM-Zeilen, die nur aus Gründen der Übersichtlichkeit eingefügt wurden, nach Fertigstellung des Programms entfernt werden. Dazu eignet sich unser BASIC-Kompressor im COMMODORE WELT Special C16/PLUS4 5/88 sehr gut.

Beispiel 1:
```
20 for i=1 to 5000:next
```
Beispiel 2:
```
20 for i=1 to 5000 30 next
```
Beispiel 3:
```
20 for i=1 to 5000 30 nexti
```
Beispiel 4:
```
20::for i=1 to 5000
30 ::::nexti
```

Die erste Schleife braucht 398 Einheiten. Werden zehn Leerzeichen eingefügt, so dauert es 496 und bei zehn Doppelpunkten sogar 895 Einheiten, also mehr als doppelt so lange. Noch langsamer wird das Ganze, wenn NEXT in einer extra Zeile steht oder anstatt NEXT NEXTI geschrieben wird. Merke: Die Schleifenvariable hinter NEXT ist nur notwendig, wenn es bei verschachtelten Schleifen Verwechslungen geben könnte.

Unnötige Sprünge vermeiden

Bei einem GOTO- oder GOSUB-Befehl wird zuerst die Sprungadresse gelesen und mit der aktuellen Zeilennummer verglichen. Ist das Sprungziel größer, wird ab der nächsten Zeile jede folgende Zeilennummer mit dem Sprungziel verglichen, bis die Zeile erreicht ist. Ist die Sprungadresse kleiner, so beginnt die Suche am Programmanfang.

PROGRAMME LANGSAMER MIT GOTO/GOSUB

Der Unterschied wirkt sich vor allem bei langen Programmen aus, bei denen bestimmte Unterprogramme oft angesprungen werden. Bei jedem GOTO oder GOSUB sollte also vorher überlegt werden, ob sich eine Abtrennung des Programmteils lohnt und wo das Unterprogramm stehen soll. Als Faustregel gilt: Unterprogramme, die nur einmal durchlaufen werden, stehen am Programmende und häufig gebrauchte Routinen am Anfang.

```
20 i=i+1:if i<2000 then 20
90 i=i+1:if i<2000 then 90
```

Zum Beispiel braucht die Programmzeile mit Nummer 20, nur 1059 Zeiteinheiten (=17,65 Sekunden), während sie mit der Nummer 90 bereits 1208 Einheiten benötigt, ein Laufunterschied von etwa 2,5 Sekunden. Hierbei ist die Länge der vor Zeile 90 liegenden BASIC-Zeilen unwesentlich. Der Interpreter liest immer nur die Koppeladressen, die anzeigen, wo die folgende Zeile beginnt, und die Zeilennummer steht. Solche Schleifen sollten ohnehin nie mit GOTO gemacht werden, eine DO/LOOP/UNTIL-Schleife eignet sich dazu besser:

```
20 do:i=i+1:loop
     until i=2000
90 do:i=i+1:loop
     until i=2000
```

Dafür werden, unabhängig von der Zeilennummer, nur 978 Einheiten gebraucht. Bei DO/LOOP- und FOR/NEXT-Schleifen wird nämlich der Beginn der Schleife beim ersten Durchlauf abgespeichert und muß daher nicht extra gesucht werden. Auch das Lesen der Zeilennummer bei GOTO und GOSUB kostet Zeit. Deshalb sind kleine Zeilennummern besser. Ein fertiges, nicht mehr zu änderndes Programm, sollte stets mit RENUMBER1,1 neu numeriert werden, oder alternative, in gleichmäßigen Zweier-, Fünfer- oder Zehner-Schritten durchnumeriert sein. Allein wegen der Übersichtlichkeit ist dies zu empfehlen.

Standard-Variablen vordefinieren

Taucht im Programm eine neue Variable auf, so wird der Variablenname vom Interpreter hinter die bereits definierten Variablen geschrieben. Wird dieser Variablen ein neuer Wert zugewiesen, muß der Name vom Beginn des Variablenspeichers an gesucht werden.

VARIABLE RICHTIG EINSETZEN

Damit die am häufigsten benötigten Variablen schnell gefunden werden, sollten sie vordefiniert werden. Dies geschieht mit der DIM-Anweisung aus dem BASIC-Wortschatz. Taucht im Programm eine Zahl häufiger auf, wird auch sie als Variable definiert.

```
10 dim a(100000), i,b$,
     c%:k=1000
20 ti$="000000"
30 fori=1to5000:b
     =1000:next
40 printti
```

In diesem Beispiel wird bei jedem Durchlauf die Zahl 1000 gelesen und interpretiert. Daher benötigt Zeile 30 ganze 1642 Einheiten. Wird in Zeile 30 die Zahl 1000 durch die Variable K ersetzt, werden nur noch 892 gebraucht. Dieser Fehler wird besonders häufig bei Spielen gemacht, bei denen in den Bildspeicher gePOKEd wird, zum Beispiel POKE 3072+40*y+x. Die Anzahl der verschiedenen Variablen sollte so klein wie möglich gehalten werden. Wird ein Wert einer Variablen nicht mehr benötigt, läßt sich der Variablen-Name wiederverwerten.

GESCHWINDIGKEITSSTEIGERUNG IM BASIC

Auch der Variablentyp spielt eine Rolle bei der Geschwindigkeitssteigerung. Zwar brauchen Arrays mit Integerzahlen (mit %) wesentlich weniger Speicherplatz. Aber wenn es darum geht, Rechenoperationen damit auszuführen, so sind normale Variablen etwas schneller, da der Computer intern immer mit Fließkommazahlen arbeitet und diese im Fall von Integerzahlen erst umrechnen muß.
Probieren Sie diese Tips einmal alle bei Ihren Programmen aus. Sie werden staunen, was sich da herausholen läßt. Sollten diese Tips für Ihr Programm noch nicht genügen, werden Sie in der nächsten SEITE 16 weitere Möglichkeiten zur Ge-

schwindigkeitssteigerung finden. □

Figuren als Strings

Bei den meisten Spielen, die mit veränderten Zeichensatz arbeiten, werden die Figuren aus mehreren Zeichen zusammengesetzt. Einige Autoren machen dabei den Fehler, diese Zeichen mit CHAR, POKE oder PRINT einzeln auszugeben. Die einfachste Methode ist die Anwendung des CHAR-Befehls. Anstatt aber untereinanderstehende Zeichen einzeln auszugeben, kann man die ganze Figur in einem einzigen String zusammenfassen. Damit dies leichter verständlich wird, haben wir ein kleines Reaktionsspiel entworfen (Würfelstop), bei dem Würfelaugen verglichen werden müssen:

STRINGS ALS ZEICHENSATZHILFE

Die Darstellung der möglichen Würfeloberflächen sind im Stringfeld W$ enthalten. Der Trick bei der Darstellung der Würfel besteht in Zeile 100 in der Definition von B$. Ein Würfel hat neun Felder, die in drei Dreier-Reihen zusammengefaßt sind. Es wird nun zunächst die erste Reihe ausgegeben, mit B$ der Cursor eine Zeile tiefer und drei Zeichen zurückgesetzt und so weiter. Tippen Sie das Spiel ein, Sie werden sehen, wie schnell das geht. Wählen Sie am Anfang eine höhere Zahl für die Geschwindigkeit zum Beispiel 50. Es erscheinen dann zwei Würfel, wobei der linke unbewegt bleibt und der rechte sich schnell verändert. Stimmen die Würfelaugen überein, muß eine Taste gedrückt werden. Ziel ist es, möglichst viele richtige Stopversuche zu erreichen und möglichst wenige Übereinstimmungen zu verpassen. □

Programmieren Sie mit LABELs

Was das Programmieren mit BASIC oft besonders unübersichtlich macht, sind die Zeilennummern bei Sprungbefehlen.
Bei einem längeren Programm kommen in der Regel viele GOTO- oder GOSUB-Befehle vor.
Diese Adressen sollten notiert werden, um Fehler auszuschließen. Nach jedem RENUMBER ändern sich diese Zeilennummern jedoch und man muß dauernd nachsehen, welche Routinen nun gerade angesprungen wird.
Besonders verhängnisvoll kann sich eine falsche Zeilennummer bei einem TRAP-Befehl auswirken. Solche Fehler sind oft schwer zu finden. Wie schön wäre es doch, wenn man statt GOTO 1020 schreiben könnte GOTO EINGABE. Mit einem einfachen Trick und einem Textverarbeitungssystem wie Script/Plus ist dies möglich. Betrachten Sie folgendes Beispielprogramm:

Beispielprogramm
zu Label-Definitionen:

```
10 trap"ENDE"
20 goto"START"
   :rem ueberspringen
   der label-definitionen
30 goto"PRG1"
   :rem unterprogramm 1
40 goto"PRG2"
   :rem unterprogramm 2
50 "START"input"1 oder
   2";g:ifg=2then"PRG2"
60 "PRG1"print"unter-
   programm 1":goto
   "ENDE"
70 "PRG2"print
   "unterprogramm 2"
80 "ENDE"end
```

Anstelle der Zeilennummern werden die Namen der Zieladressen geschrieben. Damit sich diese vom Rest des Programms deutlich abheben, werden sie in Anführungszeichen gesetzt und groß geschrieben. Die Zeile, die von GOTO... angesprungen werden soll, beginnt ebenfalls mit diesem Namen. Außer der besseren Übersichtlichkeit hat dieses Verfahren einen weiteren Vorteil. Ein RENUMBER kann problemlos durchgeführt werden und geschieht blitzschnell.

LABELS STATT GOTO UND GOSUB

Natürlich ist ein solches Programm nicht lauffähig, weil Sie keinen Amiga besitzen, der dies mit seinem BASIC kann.
Nach der Fertigstellung müssen die LABELs wieder in Zahlenwerte umgesetzt werden. Ein entsprechendes Programm ist in Vorbereitung. Die Umsetzung kann mit wenig Mühe jedoch auch mit Script/Plus geschehen. Dazu wird das Programm als SEQ-File abgespeichert. Dieses File wird mit Script/Plus wieder eingeladen, die Namen werden mit dem SEARCH & REPLACE-Befehl durch die Zeilennummern ersetzt.

```
open8,8,8,"name,s,w"
:cmd8:list:close8
```
□

SEQ-FILEs in Programme umwandeln

Um die sequentiellen Files wieder in lauffähige Programme umzuwandeln, haben wir das Programm 'SEQ-TO-PROGRAMM' entwickelt:

(SEQ) Sequentielle Programm-Files umwandeln in lauffähige (PRG) Programm-Files:

```
62999 rem *** seq-to-
      programm***
63000 poke55,1:poke56,
      80:clr:input
      "programmname";
      p$
63010 open2,8,2,p$+
      ",s,r"
      :rem kanal oeffnen
      zum seq-file lesen
63020 fori=20500to
      50000:get#2,a$:
      pokei,asc(a$):ifst
      =0thennext
63030 pokei+1,255:pokei
      +2,13:close2:i=
      20500
63040 printchr$(147);
      :rem bildschirm
      clr-kode
63050 a=peek(i):print
      chr$(a);
63060 ifa=13then63080
      :rem return-kode
63070 i=i+1:goto63050
63080 key1,chr$(019)
      +chr$(013)+"i="
      +str$(i+1)+":ifer
      <>11g063040"
      +chr$(013)
63090 poke2035,0:
      sys56364
      :rem f1 taste
      aktivieren
```

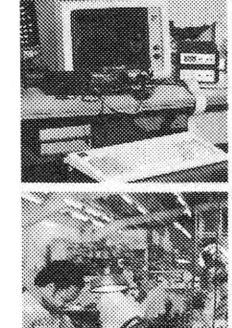
Mit Hilfe dieses Programms ist es auch möglich, ein Programm in Script/Plus zu schreiben, und in ein normales Programm umzuwandeln. Professionelle Werkzeugmaschinen können mit diesem Trick programmiert werden, wie folgt: Befehlscode im Textverarbeitungsprogramm eingeben, mit allen Vorteilen die ein solches Programm bieten kann, suchen – ersetzen – nachladen – korrigieren – kopieren – einfügen – wiederholen – löschen – kontrollieren etc. Diese Methode ist der Eingabe mit einem Ein-Zeilen-Editor mit Zahlencodeeingabe um einiges überlegen. Dann über RS 232-Schnittstelle die Daten in original ASCII-Code in den RAM-Speicher der Werkzeugmaschine schreiben und los geht die Produktion. Der kürzeste Weg ist oft der optimale.

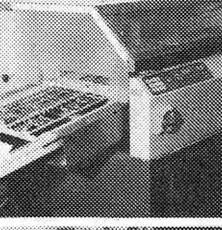
Rudolf Schmid-Fabian □

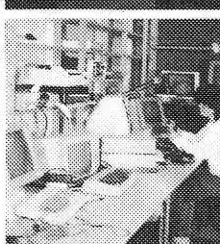
Rund um C 16 & P 4

An dieser Stelle werden wir Hard- und Software-Produkte für den C16/C116/Plus4 vorstellen und erläutern, wenn möglich, die Bezugsadressen nennen oder auf den Hersteller verweisen. Wenn nichts anderes angegeben ist, vergessen Sie bitte nicht, bei Anfragen einen frankierten Rückumschlag beizulegen.

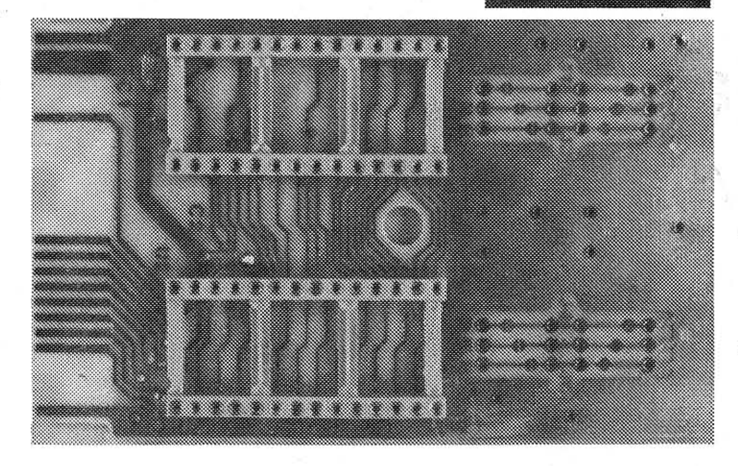












EPROMPLATINE FÜR C16/PLUS4

Ab sofort wird eine Epromplatine für den C16 beziehungsweise Plus4 angeboten. Die Platine ist mit zwei hochwertigen Präzisionssokkeln ausgestattet und ermöglicht die Benutzung von ein oder zwei Eprom der Typen 2764 oder 27128.
Eine Beschreibung erklärt die Anwendung der Platine. Der Preis für dieses Zubehör liegt inklusive Original-Commodore-Gehäuse bei 18 Mark. Wir meinen, ein akzeptierbarer Preis für diese professionell gefertigte Karte mit vergoldeten Kontakten. □

NEUER KATALOG

Die Firma Wiesemann & Theis GmbH Microcomputertechnik, Hersteller und Vertreiber diverser Interfaces, Buffer, Leistungstreiber, T-Switches, Kabel-, Video-, Tastatur- und Display-Produkten, hat einen neuen vierfarbigen Produktekatalog aufgelegt.
Ein Beispiel: Ein Hardware-Interface/Buffer ermöglicht den Anschluß eines Gerätes (Drucker, IBM PC etc.) mit V.24 (RS232)-Schnittstelle an die serielle Commodore-Bus-Schnittstelle (C64/128PC/VC20/C16/C116/Plus4).
Der Katalog ist kostenlos zu beziehen von:
Wiesemann & Theis GmbH
Abteilung: CW
Winchenbachstraße 3–5,
D-5600 Wuppertal 2. □

PLUS4-TASTATUR AN C116

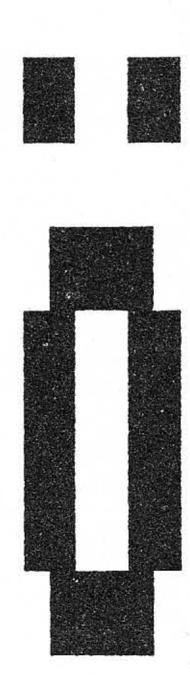
Ein Restposten von Plus4-Tastaturen ist im Elektronikfachhandel aufgetaucht. Wer sich immer noch mit der Gummitastatur des C116 herumärgert oder wegen ihr den C116 in die Ecke gestellt hat, kann den Rechner wieder reaktivieren. Für etwa 30 Mark ist eine Plus4-Tastatur zu erwerben. Vier Gehäuseschrauben öffnen, Tastaturstecker abziehen, Gummitastatur entfernen, neue Tastatur einstecken und los geht die Eingabe. Zwar ist die Plus4-Tastatur etwas größer als das gesamte Gehäuse des C116 und kann nicht eingebaut werden. Aber ein geschickter Bastler kann sie und die Platine separat auf Eigenkonstruktionen unterbringen. Die Tastatur ist auch als Ersatz für eine defekte Plus4-Tastatur zu verwenden. Zusätzlich zur Tastatur erhält man eine VC20-Platine, die als Druckpuffer oder Druckertreiber umgebaut werden kann: Eingabe über eine Plus4-Tastatur, Anpassung möglich. Info auf Anfrage von CW. □

64 KBYTE RAM FÜR C16/C116

Wer sich immer noch mit den 12277 BYTES FREE herumschlägt, dem sei eine RAM-Erweiterung nahegelegt. Die dann erscheinende Systemmeldung 60671 BYTES FREE ermöglicht auch die Eingabe von Programmen, die dem Plus4 vorbehalten waren. Dieses Steckmodul für den Expansionsport läuft beim größten Teil der C16-Rechner ohne weiteren Eingriff in den Computer.
Bei einem anderen Teil der Baureihe C16 muß das interne RAM ausgeschaltet werden. Dies erfordert einen kleinen Eingriff in den Rechner, eine Leiterbahn muß durchtrennt werden.
Eine Anleitung zur Nutzung der Erweiterung liegt bei. Der Preis von 39,90 Mark entspricht der Leistung der 64-KByte-Erweiterung. Der neue Katalog „Volles Programm" ist anzufordern bei:
DELA
ELEKTRONIK GMBH
Abteilung: CW
Maastrichter Straße 23,
D-5000 Köln 1. □

ZEICHENSATZ-ANPASSUNG

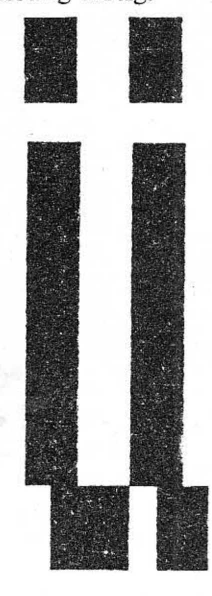
Deutscher Zeichensatz: Kein Problem für Ihren C16 / P4



Unsere Zeichensatzprogramme wurden neu überarbeitet. Statt vieler Einzelprogramme gibt es jetzt nur mehr zwei Teilprogramme. Als Ergänzung ist nun auch die Ausgabe mit dem Plus4 über Centronicskabel möglich. Ein Komprimierungsprogramm sorgt für superkurze Ladezeiten. Anpassungsvorschläge finden sich in Hülle und Fülle.

DIE EINHEIT IN DER DREIHEIT

Das Programm besteht nun aus zwei Teilprogrammen. Programmteil drei läßt sich auch in Verbindung mit unseren früheren Zeichensatzprogrammen nutzen. Es vereint die einzelnen Teile zu einem einzigen Programm, das wesentlich kürzer als einer der ursprünglichen Teile ist. Klingt ein wenig mystisch, nicht wahr? Das Geheimnis dabei ist, Teil eins und zwei sind Basic-Programme, die die verschiedensten Anpassungsarten zulassen. Teil eins mit Namen "SCREEN/KEYBOARD" ermöglicht die Definition eines neuen Bildschirmzeichensatzes und die Änderung der Tastaturbelegung. Beides ist nun unabhängig voneinander, weswegen zur Umbelegung der Tastatur im Gegensatz zu vorher nicht noch unnötigerweise die Eintragung der Zeichenmatrix vorgenommen werden muß, oder umgekehrt. Die Verlegung des Basic-Endes und somit der Schutz des neuen Bildschirmzeichensatzes erfolgt nun nicht fehlerhafterweise erst am Ende des Programmes, wo es schon längst zu spät sein kann und die Basic-Variablen dem Zeichensatz unter Umständen schon den Garaus gemacht haben können, sondern bereits am Programmanfang. Eine weitere Besonderheit ist, daß die für den Umtausch der im Tastaturpuffer liegenden Zeichen verantwortliche Maschinenroutine nicht fest im Stack lokalisiert ist, sondern durch eine einfache Änderung der Routinenadresse nach Belieben im ungebankten RAM-Bereich verschoben werden kann. Das gleiche gilt auch für den Programmteil Nummer zwei, bei welchem gleich drei Blöcke von Maschinenroutinen vorliegen. Das sind die Druckerausgaberoutine mit eventueller ASCII-Wandlung, die Centronics-Ausgabe, und die Routine, die die Einbindung der Druckerausgabe in die integrierte Software des Plus4 besorgt. Teil zwei die „Druckeranpassung“ beinhaltet so die verschiedensten Anpassungsmöglichkeiten, wie zum Beispiel die Auswahl, ob serielle Ausgabe oder Centronics, ob Commodore-Code oder ASCII, ob integrierte Plus4-Software oder normal, und ob eine Änderung der Geräte- und der Sekundäradresse vorgenommen werden soll. Anstatt bei der Druckerausgabe nur den Ausgabecode in einen anderen zu wandeln, kann eine ganze Code-Sequenz an den Drucker ausgegeben werden, wodurch auch beliebige Zeichen mittels direkter Nadelansteuerung frei definierbar sind, und somit auch grafikfähige Drucker, die keinen deutschen Zeichensatz haben, auf diese Weise einen solchen verpaßt bekommen können. Teil drei erzeugt ein viertes Programm, das sich aus den bereits fertig installierten Maschinenroutinen und Datenteilen der ersten beiden Programme zusammensetzt. Da hier keine Anpassungen mehr zu erfolgen brauchen, kann dieses Programm sehr kurz ausfallen. Datassettenbenutzer werden ihre helle Freude daran haben. Das Programm schreibt die erfaßten Daten einfach wieder in die richtigen Speicheradressen und schon ist die Zeichensatz- und Druckeranpassung fertig.

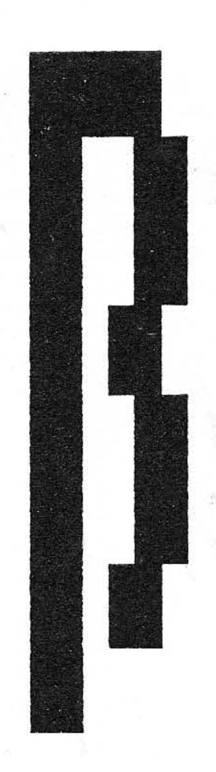


„SCREEN/KEYBOARD", DIE BILDSCHIRM- UND TASTATURANPASSUNG

Wenn Sie Zeile 125 so lassen, wie sie ist, kommt die Tastaturroutine in den Stack zu liegen. Erfahrene Maschinenprogrammierer können diese Adresse ihren Bedürfnissen gemäß verändern. In Zeile 150 steht die Adresse des neuen Zeichensatzes. Es sollten hier nur gerade Zahlen Verwendung finden. Für eigene Programme ist der zur Verfügung stehende Speicherplatz abzüglich zwei KByte zu empfehlen. Für einen Hauptspeicher von 64 KByte allerdings darf nur eine 60 angegeben werden, da sonst der Zeichensatz in die I/O-Adressen käme und so einen Systemabsturz herbeiführen würde. Es gilt also folgende Tabelle:

Hauptspeicher	Zeichensatz (in KByte)
16	14
32	30
64	60

Für die integrierte Software des Plus4 nehmen Sie bitte auch 60 KByte als Zeichensatzadresse.
Bei Script-Plus sollten Sie folgendes wissen. Eigentlich ist es beinahe egal, wo der Zeichensatz steht. Er darf aber nicht im gebankten RAM-Bereich ab 32 KByte stehen, und darf auch nicht mit unseren Anpassungsprogrammen kollidieren. Script-Plus fragt den TED-Chip nach der Lage des Zeichensatzes ab, kopiert ihn an den Anfang des zur Verfügung stehenden Speichers und stellt den TED-Chip dementsprechend um. So können Sie ruhigen Gewissens 14 KByte als Zeichensatzadresse nehmen. Wichtig ist, sofern Sie einen Rechner mit nur 16 KByte Hauptspeicher Ihr eigen nennen, daß Sie ihn durch das Verlegen des Basic-Endes mit einem entsprechenden Eintrag in Zeile 170 vor dem Überschreiben durch Basic-Variablen schützen. Wenn Sie allerdings die Zeichensatzadresse 12 KByte wählen, ist letzeres nicht mehr nötig, da erfahrungsgemäß die Variablen nicht bis an Ihren Zeichensatz kommen. Sollte Ihnen Ihr altes Zeichensatzprogramm auf Ihrem nicht erweiterten C16/C116 Schwierigkeiten gemacht haben, so können Sie diese durch eine Zwölf in Zeile 150 beseitigen.
Normalerweise sollte durch ein "J" in Zeile 170 stets das Basic-Ende verlegt werden. Wenn Sie allerdings gerne den Basic-Anfang hochsetzen und Ihren Zeichensatz lieber am Anfang ab $1000 (4096) hätten, so steht Ihnen diese Möglichkeit durch ein "N" in Zeile 170 offen.
Kommen wir nun zu einem Kernpunkt des Anpassungsprogrammes, die eigentliche Bildschirmanpassung. Dort, wo früher Buchstaben standen, haben wir nun aus Gründen besserer Lesbarkeit statt dessen die ASCII-Werte der Zeichen erfaßt, und das Ende dieser Tabelle mit einer "–1" markiert. Diese Tabelle in den Zeilen 190 und 195 korrespondiert mit den Daten der Zeichenmuster in den Zeilen 210 bis 285. Die ASCII-Werte in der ersten Tabelle haben noch nicht unbedingt etwas mit den tatsächlichen Tasten, auf denen später einmal diese Zeichen erscheinen sollen, zu tun, wohl aber mit den Codes, unter denen diese im Hauptspeicher des Rechners erfaßt werden sollen. Auch die Werte, die der Drucker später einmal bekommt, können davon unabhängig sein, sofern mit dem Programm „Druckeranpassung" dafür Sorge getragen wird. Wichtig ist nur, daß der Rechner, beziehungsweise das Verwendung findende Programm, wenn vielleicht nicht die vorgesehenen ASCII-Codes,



so doch wenigstens diejenigen, die dasselbe Zeichenmuster erzeugen, auf den Bildschirm ausgeben läßt. So sind zum Beispiel Codes 192 bis 223 identisch mit 96 bis 127, Codes 224 bis 254 identisch mit 160 bis 190, und Code 255 identisch mit 126. Aus Gründen besserer Übersichtlichkeit ist aber doch der ASCII-Wert zu wählen, der beim Abfragen der entsprechenden Taste mit '10 GETKEYX$:PRINTASC(X$)' auf dem Bildschirm erscheint. Welche Codes wir verwenden können, hängt, wie gesagt, von der zur Anwendung kommenden Software ab. In eigenen Programmen sind Sie ziemlich frei in der Wahl, auf welche Zeichenmuster Sie anstelle der deutschen Umlaute verzichten wollen. Es empfiehlt sich aber, wenn Sie bereits einen Commodore-Drucker mit deutschem Zeichensatz Ihr eigen nennen, diejenigen ASCII-Werte zu verwenden, die bei Ihrem Drucker bereits den Ausdruck des entsprechenden Zeichens bewirken.
So können Sie zum Beispiel mit der integrierten Software auf das zweite Programm, die „Druckeranpassung" völlig verzichten. Für „äöüß" können Sie die Werte 251 bis 254 oder 187 bis 190 verwenden, für „ÄÖÜ" die Werte 219 bis 221 oder 123 bis 125, für „c" stehen Ihnen gar drei Werte zur Auswahl, nämlich 126, 222 und 255. Für Script-Plus steht die Sache anders. Da dieses Programm fast alle unsere schönen Commodore-Grafikzeichen eliminiert, müssen wir uns auf die uns verbleibenden beschränken. In den Zeilen 190 und 195 sehen Sie die von uns ausfindig gemachten ASCII-Werte, die sich leider völlig von denen, die an den Drucker auszugeben sind, unterscheiden, die „Druckeranpassung" wird in diesem Falle daher zum unbedingten Muß. Die ASCII-Werte in den Zeilen 190 und 195 korrespondieren mit der Zeichenmatrix-Tabelle in den Zeilen 210 bis 285 in folgender Weise. Dem ersten ASCII-Wert ist die erste Zeichenmatrix zugeordnet, dem zweiten die zweite und so fort. Jede Zeichenmatrix besteht aus acht Data , die, binär gewandelt, das Zeichenmuster sichtbar werden lassen.
Da Sie vielleicht weitere Zeichenmuster benötigen, geben wir Ihnen diese an die Hand.

Weitere Zeichenmuster:

60, 102, 110, 110, 96, 98, 60, 0	(Klammeraffe)
12, 18, 48, 124, 48, 98, 252, 0	(engl. Pfund)
60, 96, 60, 102, 60, 6, 60, 0	(Paragraph)

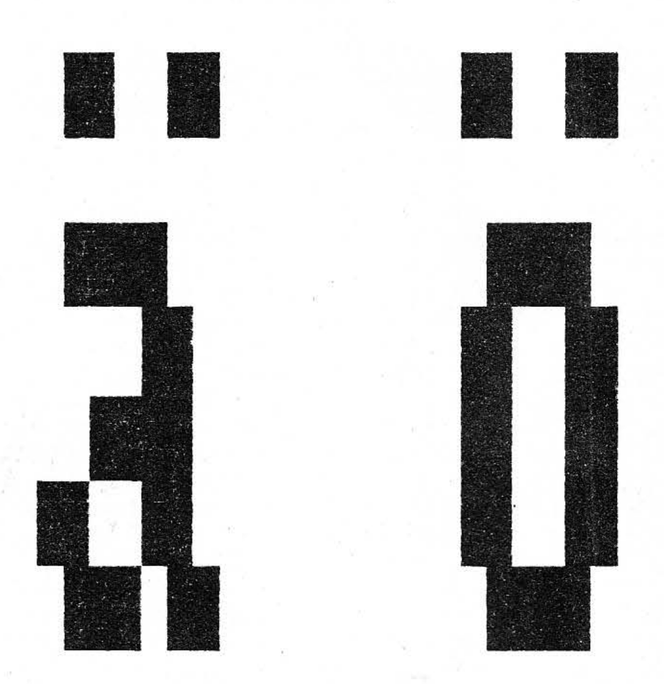
Wir hoffen, Ihnen damit geholfen zu haben, für den Fall, daß Sie auch gerne das Paragraphenzeichen verwenden möchte, oder auf den Klammeraffen und das Pfundzeichen nicht gerne verzichten möchten.

TASTATURANPASSUNG

Nun wird es richtig ernst. Mit der Tastaturanpassung bestimmen wir, welcher Code bei einem bestimmten Tastendruck wirklich in den Rechner wandert. Wir müssen nun genau wissen, welche Codes die verwendete Software annimmt. Wir vermögen aber auch unsere Tastatur mit verhältniswenig Aufwand so umzubelegen, daß wir plötzlich eine DIN-Tastatur vor uns haben. Betrachten wir zunächst einmal, welche Hindernisse uns die integrierte Software in den Weg legt, und woran auch Kingsoft mit seinem deutschen Zeichensatz gescheitert ist. Wenn wir jetzt nicht die Codes 251 bis 254 für die kleinen deutschen Umlaute verwenden, so erwartet uns das gleiche Schicksal. Auch die integrierte Software vollführt intern eine Zeichenwandlung. Statt den, soeben angegebenen Zeichen, stehen darauf plötzlich um 64 verminderte Werte im Rechner, die jedoch ebenfalls den Ausdruck von „äöüß" bewirken. Hätten wir jedoch gleich die verminderten Werte gewählt, so wären plötzlich irgendwelche Grafikzeichen oder sonstiges auf dem Bildschirm oder Drucker die Folge, denn die verminderten Werte erfahren eine andere Umwandlung. Dies zu wissen, ist wichtig für die „Druckeranpassung", bei der wir uns ansonsten wundern müßten, weshalb wir die eingegebenen Zeichen plötzlich nicht mehr zu finden vermögen. Für „ÄÖÜ" verwenden wir die Codes 219 bis 221, das „c" brauchen wir nicht anzupassen, sofern wir nicht die Taste umbelegen wollen. Für Script-Plus nehmen wir die bereits in den Zeilen 190 und 195 vorgestellten Codes, wobei wir ebenso auf das „c" verzichten können. Es bleibt nun nur die Qual der Wahl, auf welche Tasten wir unsere Umlaute legen möchten. Zu diesem Zweck gibt es wieder zwei korrespondierende Tabellen. In die Zeilen 315 bis 320 tragen wir den ASCII-Wert der gedrückten Taste ein, in die Zeilen 330 bis 355 den ASCII-Wert, der daraufhin anstatt dessen in den Rechner gelangen soll, sofern die Software mitspielt. Wenn Sie im Rechnerhandbuch die Tabelle mit den ASC- und CHR$-Codes aufschlagen, so können Sie die von uns gewählte Belegung ablesen. Mit 'PRINT CHR$(X)', wobei X der gewählte Wert ist, können Sie dieses aber bei nicht installiertem Zeichensatz auch auf dem Bildschirm ersehen. Im Anhang dieses Artikels bringen wir weitere Anpassungsvorschläge.

DIE DRUCKERANPASSUNG

Sei es, daß Ihr Drucker andere Codes braucht, als die Software verkraftet, sei es, daß Sie unter anderer Geräte- und Sekundäradresse den Drucker ansprechen müssen, sei es, daß Sie einen Centronics-Drucker haben und diesen nicht über ein Interface, sondern am Plus4 mit dem in der CW 12/87 erstmalig vorgestellten Centroncs-Kabel betreiben wollen, die „Druckeranpassung" macht es möglich.
Die drei Blöcke von Maschinenroutinen sind auch hier frei verschiebbar. In den Zeilen 150 und 160 finden Sie die Anfangsadressen der Ausgaberoutine und der Centronicsausgabe. Letztere beinhaltet daneben noch die Vertauschung der Geräte- und Sekundäradresse für die serielle Ausgabe, so daß Sie auf diesen Block für diesen speziellen Fall nicht verzichten können. In Zeile 1710 wird durch 'IS=818' die Lage der für die Einbindung der Druckerausgabe in die integrierte Software des Plus4 verantwortlichen Routine bestimmt. Keine Angst, wenn diese im Kassettenpuffer liegt und Sie mit Turbo-Plus die integrierte Software nützen wollen. Der Kassettenpuffer steht nach dem Aufruf der Software wieder zu Ihrer Verfügung. Etwas anderes dagegen bleibt zu beachten. Turbo-Plus benötigt in Verbindung mit der Plus4-Software einen Teil des Platzes, in der die Druckerroutine liegt. Erhöhen Sie daher in diesem Falle in Zeile 150 die Anfangsadresse der Druckerroutine um ca. 60. Es könnte auch ein bißchen weniger genügen, aber auf das Byte genau haben wir dieses noch nicht ausgemessen. Die Auswahl in den Zeilen 200, 240 und 280 dürfte wohl keiner weiteren Erläuterungen bedürfen. Die Zeilen 300 bis 370 sind für Centronics ohne Be-



lang. Erfolgt eine Abänderung der Geräteadresse, so ist auch die Angabe einer Sekundäradresse oder der Eintrag 255 erforderlich. Die Zeilen 390 bis 440 sind für den nicht völlig auszuschließenden Fall vorgesehen, daß eine abweichende Version der Plus4-Software existieren könnte, die hier eine Änderung erforderlich macht. Das Kernstück der „Druckerausgabe" bilden wieder zwei korrespondierende Tabellen. In den Zeilen 480 und 490 finden Sie die auszutauschenden Codes. Für Script-Plus sind die bereits aus der Bildschirm- und Tastaturanpassung bekannten Codes hier einzusetzen, in den Zeilen 550 bis 651 die entsprechenden Sequenzen für einen IBM-Drucker. Eine genauere Erläuterung finden Sie im Anhang des Artikels.

ZUM PROGRAMMABLAUF

Folgendes ereignet sich beim Start von "SCREEN/KEYBOARD". Der Rechner besorgt sich die Adresse

des neuen Zeichensatzes und überprüft, ob eine Verlegung des Basic-Endes stattfinden soll. Ist dieses der Fall, so findet ein POKE in die Speicherstellen 55 und 56 statt. Ein CLR besorgt den Rest. Die vordem gelesenen Werte werden erneut eingelesen und dann geht es an die Installierung der Zeichensatzverschiebung. Diese Routine kommt nun an die in der Zeile 125 angegebene Stelle, wird dementsprechend angepaßt und mit der Adresse des neuen Zeichensatzes versehen. Nach dem SYS-Aufruf erfolgt die Verschiebung und der Einbau der in den Zeilen 210 bis 285 definierten Zeichenmuster an die durch die Zeilen 190 bis 195 festgelegten Stellen. Die zwei durch die Zeilen 315 bis 320 und 350 bis 355 bestimmten Tabellen werden nun an den dafür vorgesehenen Platz gePOKEd und die Tastaturroutine überschreibt nun die nicht mehr benötigte Verschieberoutine des Zeichensatzes. Nachdem sie an ihre Lage und an die der zwei Tastaturtabellen angepaßt ist, kann die Umstellung des TED-Chips auf die neue Zeichensatzadresse und die Einbindung der Tastaturroutine in den Systeminterrupt erfolgen.

PROGRAMMABLAUF „DRUCKERANPASSUNG"

Nach dem Einlesen der Parameter bis einschließlich Zeile 440, beginnt die Kreierung der zwei Tabellen mit den zu tauschenden Codes und den dafür zu sendenden Sequenzen. Wenn diese im Hauptspeicher stehen, folgt das POKEn der Druckerroutine und deren Anpassung an ihre Lage. Ob es eine Centronics-Routine braucht oder nicht, es wird nicht lange gefackelt und dieselbe ebenfalls dorthin, wo sie soll, abgelegt. Ist serielle Übertragung gefragt, so werden die entsprechenden Sprünge in die Druckerroutine eingetragen. Unterbunden wird die Verknüpfung mit dem ASCII-Wandler, sollte Commodore-Code statt ASCII Trumpf sein. Nun kommt die Zeit, an die Verknüpfung der Druckerroutine mit den Tabellen der auszutauschenden Codes zu denken. Was noch bleibt, ist die Einbindung der neuen Routinen in die Systemroutinen BSOUT, OPEN, CLOSE und CLRCH durch eine geeignete Umstellung der im Bereich $300 bis $330 (768 bis 816) angesiedelten Sprungvektoren. Wenn eventuell noch nötige Einbinderoutine für die Plus4-Software installiert ist, so ist das Werk endlich vollbracht.

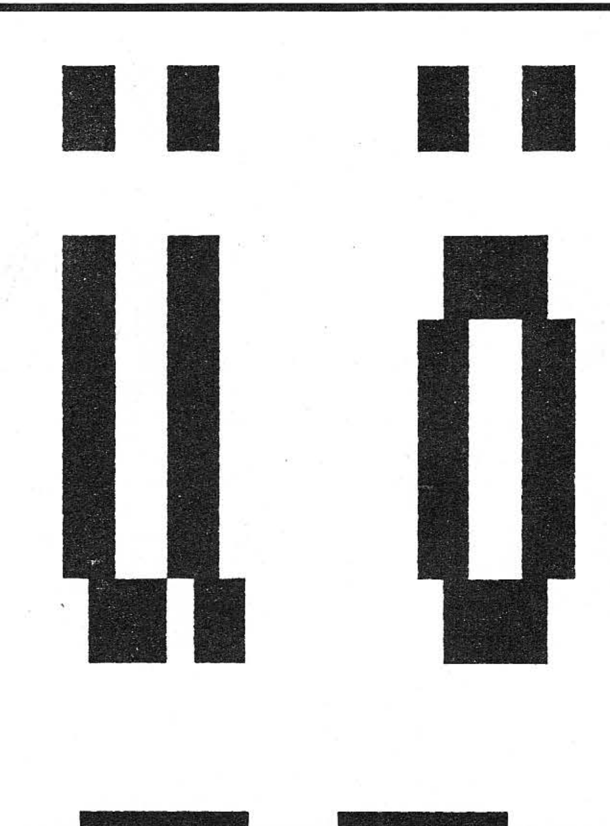
ZEICHENSATZKOMPRIMIERUNG

Nachdem Ihr Zeichensatz installiert ist, und Sie dessen Funktionieren überprüft haben, können Sie eine Programmkomprimierung vornehmen. Sie laden hierzu das Programm "ZSKOMPRIMIERER", das eventuell noch an den verwendeten Zeichensatz anzupassen ist. In die Zeilen 115 bis 120 sind die ASCII-Werte der veränderten Bildschirmzeichen einzutragen, damit das Programm sich deren Daten zu besorgen vermag. Sollten Sie die Anfangsadressen einiger Maschinenroutinen verschoben haben, so können Sie dieses entsprechend in den Zeilen 135 bis 150 vermerken. Wenn einige der dort angegebenen Routinen in Ihrer Anpassung keine Verwendung finden, so steht es Ihnen frei, eine weitere Verkürzung vorzunehmen. Nach der jeweiligen Anfangsadresse ist die Anzahl der zu übertragenden Bytes angegeben. Sie können sie auf eins herabsetzen, wenn die Routine nicht benötigt

Lesen Sie bitte weiter auf Seite 30

```
55 rem screen/keyboard=========c16 <hp>
60 rem (p) commodore welt team      <fh>
65 rem ============================ <of>
70 rem (c) by alfons mittelmeyer    <bj>
75 rem                              <cf>
80 rem                              <cp>
85 rem                              <dj>
90 rem basic v3.5                   <im>
95 rem c16/116/plus4                <gl>
100 rem ========================== <id>
105 rem    zeichensatzanpassung    <eh>
110 rem ========================== <ob>
115 rem  programmanfang            <gf>
120 rem  (verschiebbar)            <il>
125 data 312:rem stack             <lb>
130 rem -------------------------- <hh>
135 rem  neue zeichensatzadresse   <fk>
140 rem  14=scriptplus             <be>
145 rem  60=plus4-software         <ac>
150 data 14:rem 14kb=$3800 (14336) <aj>
155 rem -------------------------- <oa>
160 rem  basicende verlegen        <mf>
165 rem  j=ja n=nein               <kp>
170 data j:rem ja                  <jl>
175 rem -------------------------- <be>
180 rem  ascii-werte der zeichen   <nn>
185 rem  (mit -1 abschliessen)     <me>
190 data 176,191,177,171,178,174   <fc>
195 data 179,222,-1:rem scriptplus <ng>
200 rem -------------------------- <bf>
205 rem  neue zeichenmuster        <dh>
210 data 000,102,060,006           <ee>
215 data 062,102,062,000:rem "ae"  <pj>
220 data 000,102,060,102           <ah>
225 data 102,102,060,000:rem "oe"  <bg>
230 data 000,102,000,102           <ej>
235 data 102,102,062,000:rem "ue"  <ic>
240 data 000,060,102,124           <oh>
245 data 102,102,124,096:rem "ss"  <hl>
250 data 102,024,060,102           <jh>
255 data 126,102,102,000:rem "Ae"  <fi>
260 data 102,060,102,102           <jk>
265 data 102,102,060,000:rem "Oe"  <lk>
270 data 102,000,102,102           <ai>
275 data 102,102,060,000:rem "Ue"  <kh>
280 data 000,000,003,062           <je>
285 data 116,054,054,000:rem [pi]  <pg>
290 rem ========================== <bk>
295 rem     tastaturanpassung      <ae>
300 rem ========================== <kn>
305 rem  auszutauschende codes     <kj>
310 rem  (mit -1 abschliessen)     <gp>
315 data 64,92,42,94,186,169,192   <nj>
320 data 164,168,223,219,255,-1    <bd>
325 rem -------------------------- <ha>
330 rem  neue codes                <jc>
335 rem  (anzahl muss auszu-       <ik>
340 rem  tauschenden codes ent-    <al>
```

```
345 rem  entsprechen)                <ah>
350 data 176,191,177,171,178,174     <eg>
355 data 179,64,92,42,94,222         <bb>
360 rem ==========================   <bn>
365 rem      installation            <an>
370 rem ==========================   <ie>
375 rem  basicende verlegen          <ne>
380 readpr                           <pi>
385 reada:readb$:ifb$="n"then410     <ap>
390 poke309,a:x=a*4:poke55,0         <pc>
395 poke56,x:clr:a=peek(309)         <ab>
400 rem --------------------------   <fc>
405 rem  zeichensatz verschieben     <gb>
410 restore:readpr                   <bk>
415 x=a*4:restore 420                <ca>
420 data 160,8,162,0,189,0,208       <ja>
425 data 157,0,48,232,208,247        <bd>
430 data 238,62,1,238,65,1,136       <me>
435 data 208,238,96                  <pj>
440 fori=prtopr+22:reada:pokei,a     <lf>
445 next:ax=pr+6:gosub725            <ib>
450 pokepr+14,lb:pokepr+15,hb        <fe>
455 ax=pr+9:gosub725                 <am>
460 pokepr+17,lb:pokepr+18,hb        <nn>
465 pokepr+9,x:syspr:b=x*256         <ki>
470 rem --------------------------   <ia>
475 rem  zeichen aendern             <lb>
480 restore:reada$                   <ih>
485 reada$:reada$:x$=""              <fa>
490 reada:ifa=-1then500              <kf>
495 x$=x$+chr$(a):goto490            <pp>
500 n=len(x$):forj=1ton              <in>
505 scnclr:printmid$(x$,j,1)         <kp>
510 a=peek(3072):ad=8*a+1024+b       <of>
515 fori=0to7:reada                  <ln>
520 pokead+i,a:next:nextj            <ap>
525 rem --------------------------   <do>
530 rem  tastaturanpassung           <kf>
535 a=0:x$=""                        <gd>
540 reada:ifa=-1then550              <hk>
545 x$=x$+chr$(a):goto540            <in>
550 n=len(x$):y$="":fori=1ton        <ap>
555 reada:y$=y$+chr$(a):next         <pi>
560 restore565                       <pj>
565 data 32,191,207,32,205,206       <ke>
570 data 165,251,72,169,0,133        <le>
575 data 251,8,88,32,17,219          <mk>
580 data 166,239,240,23,189,38       <nl>
585 data5,162,10,221,111,1,240       <jp>
590 data 5,202,208,248,240,8         <mk>
595 data 189,121,1,166,239,157       <ig>
600 data 38,5,169,251,45,18          <al>
605 data 255,141,18,255,76,84        <hd>
610 data 206,120,169,56,141,18       <db>
615 data3,169,1,141,19,3,88,96       <jo>
620 fori=prtopr+68                   <me>
625 reada:pokei,a:next:c=pr+55       <mi>
630 ax=pr:gosub725                   <ha>
635 pokepr+58,lb:pokepr+63,hb        <gc>
640 rem --------------------------   <hh>
645 rem  zeichensatzinstallierung    <ci>
650 rem  und einbindung der          <nd>
655 rem  tastaturroutine             <jd>
660 ad=65299:sysc+1                  <jl>
665 pokead,(peek(ad)and3)orx         <hh>
670 ax=c:gosub725:pokepr+28,lb       <nm>
675 pokepr+29,hb:pokepr+26,n         <ee>
680 ax=c+n:gosub725:pokepr+38,lb     <lb>
685 pokepr+39,hb                     <oi>
690 fori=1ton:a$=mid$(x$,i,1)        <bk>
695 pokec+i,asc(a$):next             <ne>
700 fori=1ton:a$=mid$(y$,i,1)        <ce>
705 pokec+n+i,asc(a$):next           <hi>
710 goto745                          <ej>
715 rem --------------------------   <cj>
720 rem  berechnung von lb und hb    <ah>
725 hb=int(ax/256):lb=ax-256*hb      <nb>
730 return                           <eh>
735 rem --------------------------   <na>
740 rem  ende                        <pc>
745 end                              <fh>
750 rem ========e=n=d============    <nk>
```





```
10 rem druckeranpassung========c16 <ih>
20 rem (p) commodore-welt-team     <ie>
30 rem =========================== <ng>
40 rem (c) by alfons mittelmeyer   <cg>
50 rem                             <pd>
60 rem                             <ah>
70 rem basic v3.5                  <nl>
80 rem c16/116/plus4               <ki>
90 rem                             <ed>
100 rem ========================== <id>
110 rem       druckeranpassung     <al>
120 rem ========================== <gn>
130 rem  programmadressen          <ep>
140 rem  (verschiebbar)            <hf>
150 data 1630:rem mc-ausgabe       <ma>
160 data 1015:rem centronics       <jj>
170 rem -------------------------- <ef>
180 rem  schnittstelle             <pg>
190 rem  s=seriell c=centronics    <ig>
200 data c:rem centronics          <co>
210 rem -------------------------- <da>
220 rem  codewahl                  <bg>
230 rem  c=cbm a=ascii             <mn>
240 data a:rem ascii               <hi>
250 rem -------------------------- <jg>
260 rem  betriebsart               <jd>
270 rem  n=normal s=software plus4 <be>
280 data n:rem normal              <ol>
290 rem -------------------------- <bn>
300 rem  ausgabegeraet             <bj>
310 data 4:rem standard            <ji>
320 rem -------------------------- <kp>
330 rem  sekundaeradresse          <ij>
340 rem  k  =keine aenderung       <be>
350 rem  255=keine sekundaeradr.   <dg>
360 rem  sonst gemaess eintrag     <jl>
370 data 255:rem keine sek.adresse <na>
380 rem -------------------------- <kn>
390 rem  fuer evtl. existierende   <eb>
400 rem  abweichende version der   <oj>
410 rem  plus4-software            <jk>
420 data 3b5d:rem ciout            <mj>
430 data 3b38:rem listen           <mm>
440 data 3b3f:rem second           <ie>
450 rem -------------------------- <ni>
460 rem  zu tauschende codes       <ee>
470 rem  (mit -1 abschliessen)     <fb>
480 data 176,191,177,171,178,174   <be>
490 data 179,17,92,222,13,-1       <ei>
500 rem -------------------------- <on>
510 rem  dafuer zu uebertragende   <ad>
520 rem  codesequenzen             <gh>
530 rem  (jede sequenz ist mit     <db>
540 rem   -1 abzuschliessen)       <ib>
550 data 132,-1:rem "ae"           <le>
560 data 148,-1:rem "oe"           <pk>
570 data 129,-1:rem "ue"           <jm>
580 data 225,-1:rem "ss"           <nl>
590 data 142,-1:rem "Ae"           <np>
600 data 153,-1:rem "Oe"           <cm>
610 data 154,-1:rem "Ue"           <jl>
620 data 27,54,-1:rem [lowercase]  <gp>
630 data 156,-1:rem [pfund]        <gk>
640 data 227,-1:rem [pi]           <pp>
650 data 27,54,13,10,-1:rem [cr]   <jf>
660 rem ========================= <nj>
670 rem parameter einlesen        <dg>
680 readar:readct                 <dk>
690 fori=1to8:readp$(i):next      <db>
700 tb=ar+95:rem codetabelle      <jd>
710 rem ------------------------- <bf>
720 rem druckercodes einlesen     <jk>
730 readx:ad=tb:do until x=-1     <gc>
740 pokead,x:ad=ad+1:readx:loop   <ne>
750 rem ------------------------- <ch>
760 rem codesequenzen einlesen    <fh>
770 n=ad-tb:zg=0:x=zg+n+1:ad=ad-1 <dk>
780 fori=1ton:y=x:zg=zg+1         <bn>
790 pokead+zg,y                   <pf>
800 x=x+1:reada                   <ec>
810 ifa<>-1thenpokead+x,a:goto800 <jd>
820 pokead+y,x-y-1:next           <bk>
830 rem ------------------------- <kp>
840 rem  mc-routine:druckerausgabe <ng>
850 fori=artoar+94:reada:pokei,a  <fm>
860 next                          <ea>
870 data 142,55,1,166,153,224,3   <ib>
880 data 208,6,166,173,224,116    <fp>
890 data 240,6,174,55,1,76,75,236 <dp>
900 data 32,122,6,174,55,1,96,162 <co>
910 data 12,240,40,221,186,6,240,9 <oe>
920 data 202,208,248,32,168,6,76  <nc>
930 data 247,3,189,189,6,170,189  <jd>
940 data 189,6,240,17,140,54,1,168 <np>
950 data 232,189,189,6,32,137,6   <il>
960 data 136,208,246,172,54,1,24  <ol>
970 data 96,170,41,127,201,65,48  <lj>
980 data 12,201,91,16,8,224,0,48  <in>
990 data 5,73,32,208,1,138,96     <gi>
1000 ax=ar+28:gosub2170           <bm>
1010 pokear+22,lb:pokear+23,hb    <kf>
1020 ax=ar+43:gosub2170           <nf>
1030 pokear+64,lb:pokear+65,hb    <ni>
1040 ax=ar+74:gosub2170           <eb>
1050 pokear+41,lb:pokear+42,hb    <bn>
1060 rem ------------------------ <no>
1070 rem  mc-routine:centronics   <kj>
1080 fori=cttoct+54:reada:pokei,a <ij>
1090 next                         <ao>
1100 data 141,16,253,169,0,141,2  <ec>
1110 data 253,169,8,141,2,253,169 <ib>
1120 data 32,44,1,253,240,251,24  <oe>
1130 data 96,166,174,224,4,208,8  <ap>
1140 data 162,3,134,174,162,116   <ka>
1150 data 134,173,32,83,239,162   <nk>
1160 data 0,134,173,96,32,12,239  <ld>
```

```
1170 data 240,246,32,93,238,144        <hl>
1180 data 241,96                       <jn>
1190 rem -------------------------     <am>
1200 rem  serielle uebertragung        <ad>
1210 ifp$(1)="c"then1350               <nf>
1220 pokear+9,240:pokear+10,10         <jp>
1230 x$="ec4b"                         <ee>
1240 ifp$(3)="s"thenx$="ffa8"          <lb>
1250 ax=dec(x$):gosub2170              <bb>
1260 pokear+44,lb:pokear+45,hb         <ai>
1270 x=val(p$(4))                      <hc>
1280 pokear+6,x:pokect+29,x            <lh>
1290 ifp$(5)="k"then1370               <ff>
1300 pokect+33,val(p$(5))              <ao>
1310 goto 1370                         <pb>
1320 rem -------------------------     <dh>
1330 rem  centronics                   <ik>
1340 ax=ct:gosub2170                   <ke>
1350 pokear+44,lb:pokear+45,hb         <hp>
1360 rem -------------------------     <db>
1370 rem  cbm-code                     <aa>
1380 if p$(2)="a"then1430              <ja>
1390 pokear+40,234:pokear+41,234       <ci>
1400 pokear+42,234                     <bf>
1410 rem -------------------------     <bg>
1420 rem  tabellenzeiger               <cm>
1430 ax=ar+94:gosub2170                <jl>
1440 pokear+33,lb:pokear+34,hb         <ln>
1450 pokear+29,n:ax=ax+n:gosub2170     <mh>
1460 pokear+47,lb:pokear+48,hb         <io>
1470 pokear+51,lb:pokear+52,hb         <jj>
1480 pokear+61,lb:pokear+62,hb         <dk>
1490 rem -------------------------     <kj>
1500 rem sprungvektoren                <gb>
1510 ifp$(1)<>"c"then1540              <nj>
1520 ax=ct:gosub2170                   <jm>
1530 pokear+44,lb:pokear+45,hb         <ih>
1540 ifp$(3)="s"then1700               <ob>
1550 ax=ar:gosub2170                   <cp>
1560 poke804,lb:poke805,hb             <en>
1570 ifp$(4)<>"4"then1600              <ka>
1580 ifp$(5)<>"k"then1600              <li>
1590 ifp$(1)<>"c"thenend               <mg>
1600 ax=ct+22:gosub2170                <nm>
1610 poke792,lb:poke793,hb             <gl>
1620 ifp$(1)<>"c"thenend               <ic>
1630 ax=ct+44:gosub2170                <kp>
1640 poke800,lb:poke801,hb             <kf>
1650 ax=ct+49:gosub2170                <mf>
1660 poke794,lb:poke795,hb             <ed>
1670 end                               <jj>
1680 rem -------------------------     <dg>
1690 rem  integrierte software         <pg>
1700 restore1730                       <bg>
1710 is=818:fori=istois+60             <jb>
1720 reada:pokei,a:next                <pd>
1730 data 72,173,94,59,201,168,208     <bg>
1740 data 47,173,95,59,201,255,208     <kl>
1750 data 40,169,0,141,94,59,169,7     <gf>
1760 data 141,95,59,169,247,141,57     <pp>
1770 data 59,169,3,141,58,59,169,6     <ck>
1780 data 141,64,59,169,4,141,65       <ho>
1790 data 59,169,75,141,36,3,169       <ec>
1800 data 236,141,37,3,104,76,75       <jn>
1810 data 236                          <bc>
1820 ax=dec(p$(6))+1:gosub2170         <ib>
1830 pokeis+2,lb:pokeis+3,hb           <na>
1840 pokeis+18,lb:pokeis+19,hb         <ee>
1850 ax=ax+1:gosub2170                 <ae>
1860 pokeis+23,lb:pokeis+24,hb         <gl>
1870 ax=dec(p$(7))+1:gosub2170         <ac>
1880 pokeis+28,lb:pokeis+29,hb         <nh>
1890 ax=ax+1:gosub2170                 <dc>
1900 pokeis+33,lb:pokeis+34,hb         <ho>
1910 ax=dec(p$(8))+1:gosub2170         <fl>
1920 pokeis+38,lb:pokeis+39,hb         <io>
1930 ax=ax+1:gosub2170                 <ag>
1940 pokeis+43,lb:pokeis+44,hb         <hj>
1950 pokear+4,174                      <mi>
1960 ax=ar:gosub2170                   <ip>
1970 pokeis+16,lb:pokeis+21,hb         <kc>
1980 ifp$(1)="s"then2030               <ej>
1990 ax=ct+20:gosub2170                <dm>
2000 pokeis+26,lb:pokeis+31,hb         <lm>
2010 pokeis+36,lb:pokeis+41,hb         <id>
2020 goto2120                          <om>
2030 restore2060                       <fm>
2040 fori=ct+36toct+48                 <dl>
2050 reada:pokei,a:next                <fk>
2060 data 76,177,255,201,255,208       <jh>
2070 data 3,76,252,225,76,147,255      <ml>
2080 ax=ct+22:gosub2170                <bk>
2090 pokeis+26,lb:pokeis+31,hb         <dd>
2100 ax=ct+39:gosub2170                <je>
2110 pokeis+36,lb:pokeis+41,hb         <nk>
2120 ax=is:gosub2170                   <oc>
2130 poke804,lb:poke805,hb             <po>
2140 end                               <ei>
2150 rem -------------------------     <aa>
2160 rem berechnung von hb und lb      <db>
2170 hb=int(ax/256):lb=ax-256*hb       <ja>
2180 return                            <kh>
2190 rem ------e-n-d-e-----------      <ho>
```

ZSKOMPRIMIERUNG

```
55 rem zskomprimierung=========c16     <cl>
60 rem (p) commodore-welt-team         <ib>
65 rem ==========================      <of>
70 rem (c) by alfons mittelmeyer       <bj>
75 rem                                 <cf>
80 rem basic v3.5                      <he>
85 rem c16/116/plus4                   <ok>
90 rem ==========================      <jg>
95 rem   anpassungsteil                <ia>
100 rem -------------------------      <op>
```

```
105 rem  ascii-werte der zeichen   <bp>
110 rem  (mit -1 abschliessen)     <pn>
115 data 176,191,177,171,178,174   <mg>
120 data 179,222,-1:rem scriptplus <ol>
125 rem -------------------------- <kn>
130 rem  programmadressen          <ep>
135 data 312,152 :rem tastatur     <ff>
140 data 1630,368:rem drucker      <al>
145 data 1015,64 :rem centronics   <lm>
150 data 818,61  :rem plus4-soft   <me>
155 rem ========================== <bb>
160 data 792,4,800,2,804,2         <gn>
165 data 65299,1,786,2:goto240     <ig>
170 rem -------------------------- <ef>
175 rem  subroutinen               <pg>
180 ax=a0:gosub205:pokesz,lb       <kb>
185 pokesz+1,hb:ax=a1:gosub205     <mo>
190 pokesz+2,lb:pokesz+3,hb        <lo>
195 ax=a2:gosub205:pokesz+4,lb     <nc>
200 pokesz+5,hb:sz=sz+6:return     <gf>
205 hb=int(ax/256):lb=ax-256*hb    <mc>
210 return                         <dd>
215 a2=dz-zw:gosub180:l2=sz:sz=l1  <mj>
220 a2=a1:a1=dz:gosub180:l1=sz     <ib>
225 sz=l2:dz=dz+a0:return          <pe>
230 rem -------------------------- <bc>
235 rem  basicende                 <lp>
240 gosub615:poke208,peek(55)      <ma>
245 poke209,peek(56):ax=li         <gj>
250 gosub205:poke55,lb:poke56,hb   <ci>
255 clr:gosub615:b=peek(65299)     <ci>
260 rem -------------------------- <ob>
265 rem  bildschirmcodes           <gm>
270 b=(b and251)*256:x$=""         <dm>
275 reada:ifa=-1then285            <jk>
280 x$=x$+chr$(a):goto275          <il>
285 n=len(x$)                      <dh>
290 rem -------------------------- <bn>
295 rem  variablen                 <fl>
300 l1=li:af=li+(e1+n)*6+2        <op>
305 la=af+78+bk:l2=la:zw=af-ba     <fd>
310 dz=la+(e2+n)*6+2               <pb>
315 rem -------------------------- <nl>
320 rem  zeichenmuster             <ek>
325 sz=l1:a0=1:a1=65343:gosub180   <dn>
330 l1=sz                          <hp>
335 sz=l2:a0=2048:a1=b:a2=53248    <eg>
340 gosub180                       <bo>
345 fori=1ton:scnclr               <oe>
350 printmid$(x$,i,1):a=peek(3072) <na>
355 a1=8*a+1024+b:a0=8             <pe>
360 gosub215:next                  <fl>
365 rem -------------------------- <cn>
370 rem  routinen                  <cn>
375 fori=1to9:reada1:reada0        <bj>
380 gosub215:next                  <gi>
385 rem -------------------------- <hh>
390 rem  basicende                 <oo>
395 pokedz,peek(208)               <ie>
400 pokedz+1,peek(209):a0=2:a1=55  <ch>
405 a2=dz-zw:gosub180:dz=dz+2      <po>
410 pokesz,0:pokesz+1,0:sz=l1      <pp>
415 ax=dz-zw:gosub205:pokedz,lb    <ld>
420 pokedz+1,hb                    <de>
425 a0=2:a1=45:a2=dz:dz=dz+2       <bg>
430 gosub180                       <mi>
435 rem -------------------------- <mp>
440 rem  programm verschieben      <jf>
445 a0=dz-af:a1=ba:a2=af           <ak>
450 gosub180                       <af>
455 rem -------------------------- <bf>
460 rem  neue listenadresse        <ec>
465 a0=1:a1=ba+bk+1:a2=dz          <li>
470 ax=la-zw:gosub205:pokedz,lb    <ke>
475 pokedz+1,hb:gosub180           <mp>
480 a1=ba+bk+5:a2=dz+1:gosub180    <cn>
485 ax=60491:gosub205:rem bsout    <fd>
490 pokedz+2,lb:pokedz+3,hb        <if>
495 a0=2:a1=804:a2=dz+2            <cl>
500 gosub180:a0=1:a1=65342         <ni>
505 gosub180:pokesz,0:pokesz+1,0   <om>
510 rem -------------------------- <mb>
515 rem  verschieberoutine         <bk>
520 restore550                     <hc>
525 fori=aftola-1:reada:pokei,a:ne
xt                                 <pk>
530 print"sys"af+bk"eingeben und"  <fe>
535 print"programm abspeichern"    <ig>
540 ax=li:gosub205:pokeaf+bk+1,lb  <ab>
545 pokeaf+bk+5,hb:end             <el>
550 data 13,16,10,0,158,52,49,49   <ck>
555 data 49,58,128,0,0,0           <ap>
560 data 169,0,133,208,169,48,133,
209                                <cc>
565 data 120,160,0,177,208,133,210
,208                               <lo>
570 data 7,200,177,208,208,3,88,96 <ep>
575 data 200,162,1,177,208,149,210
,200                               <ab>
580 data 232,224,6,208,246,152,72,
166                                <ei>
585 data 211,240,16,160,0,177,214,
145                                <ke>
590 data 212,200,208,249,230,213,2
30,215                             <md>
595 data 202,208,242,166,210,240,1
0,160                              <ol>
600 data 0,177,214,145,212,200,202
,208                               <od>
605 data 248,104,168,24,144,189    <oj>
610 rem -------------------------- <op>
615 li=8960:e1=16:e2=11:ba=4097    <ob>
620 bk=14:return                   <hi>
625 rem =====e=n=d=============== <ao>
```

ä ö ü - kein Problem

Fortsetzung von Seite 25

wird. Bitte tragen Sie aber keine Null ein, da die Null als Endmarkierung für die Datenübertragung fungiert. Nachdem das Komprimierprogramm mit RUN gestartet ist, überträgt es einige Daten in dafür vorgesehene Speicherplätze und trägt die Adressen weiterer Daten in zwei Listen ein. Nachdem die Verschieberoutine sich als Maschinenroutine im Rechner befindet, starten Sie diese bitte mit dem auf dem Bildschirm angezeigten SYS-Aufruf. Die Maschinenroutine arbeitet nun eine Liste von Adressen durch und besorgt sich die gewünschten Daten. Im Anschluß darauf kopiert sie sich selbst an den Basic-Anfang und stellt einen Zeiger auf die zweite Liste um, die für das Zurückschreiben der Daten verantwortlich ist. Da die Kopie über einen Basic-Programmkopf verfügt und eine entsprechende Einstellung des Variablenanfanges vorgenommen wurde, können Sie das erzeugte Programm nun wie ein normales Basic-Programm speichern, laden und starten. Nachdem Sie das Programm gespeichert haben, sollten Sie sich von dessen Funktionieren überzeugen, indem Sie für einige Sekunden den Rechner laden und starten. Sie werden sich wundern, wie schnell nun der Zeichensatz installiert ist.

Zeichensatzanpassungen:

INTEGRIERTE SOFTWARE DES PLUS4

Im Programmteil SCREEN-KEYBOARD sind folgende Zeilen abzuändern:

```
150 data 60
190 data 251,252,253,254,219,220
195 data 221,222,-1
350 data 251,252,253,254,219,220
355 data 221,64,92,94,222
```

Es ist vielleicht empfehlenswert, den Klammeraffen durch das Paragraphenzeichen zu ersetzen, indem in Zeile 195 nach der 222 noch der Wert 64 eingeschoben, das uns entsprechende Zeichenmuster in die Zeile 286 eingetragen wird. Das Zeichenmuster wurde im vorangegangenen Artikel bereits angegeben.

PLUS4-SOFTWARE MIT COMMODORE-DRUCKER

Wer einen Commodore-Drucker mit deutschem Zeichensatz sein eigen nennt, kann auf das Programm DRUCKER-ANPASSUNG verzichten. Wenn er aber dabei nicht das englische Pfund-Zeichen verlieren will, so empfiehlt es sich, in Zeile 195 noch die Zahl 237 unterzubringen und das Zeichenmuster in der bereits schon beim Klammeraffen angesprochenen Weise einzutragen.

PLUS4-SOFTWARE MIT DRUCKERANPASSUNG

Wer eine Druckeranpassung vornimmt, braucht sich in SCREEN-KEYBOARD um das Pfund-Zeichen nicht zu kümmern.
Notwendig ist hier die Abänderung der Zeilen:

```
480 data 187,188,189,190,219,220
490 data 221,17,92,222,13,-1
```

VERSCHIEDENE DRUCKERANPASSUNGEN

Ob Script-Plus oder integrierte Software, wir müssen die Code-Sequenz angeben, die unser Drucker benötigt, wenn er ein bestimmtes Zeichen darstellen soll.

COMMODORE-DRUCKER MIT DEUTSCHEN ZEICHEN

Die Codes sind uns bereits bekannt. Wir ändern im Programm DRUCKER-ANPASSUNG folgende Zeilen:

```
550 data 187,-1        610 data 221,-1
560 data 188,-1        620 data 17,-1
570 data 189,-1        630 data 237,-1
580 data 190,-1        640 data 222,-1
590 data 219,-1        650 data 13,-1
600 data 220,-1
```

IBM-DRUCKER

Die entsprechenden DATA stehen bereits im Programm. Wir wollen kurz unsere Anpassung erläutern. In Zeile 620 haben wir den ASCII-Code 17 unterdrückt, da, anstatt wie der Commodore-Drucker auf Kleinbuchstaben zu schalten, der IBM-Drucker in irgendeinen Grafik-Modus schalten und abstürzen würde. Wir haben statt dessen die 17 durch die Sequenz ESC 6 ersetzt, womit der internationale Zeichensatz eingeschaltet wird.
Script-Plus, das im Commodore-Modus am Anfang jeder Zeile mit Code 17 in den Kleinschreibmodus umschalten will, aktiviert dadurch die internationalen Sonderzeichen, unter denen sich auch unsere deutschen Umlaute befinden. Damit wir auch bei Benutzung anderer Software nicht unbedingt ESC 6 oder den ASCII-Code 17 senden müssen, haben wir in Zeile 650 den Return-Code ebenfalls mit der Zeichensatzumschaltung versehen. Spätestens in der zweiten Ausgabezeile ist der deutsche Zeichensatz aktiv.
Vollzieht der Drucker nicht bereits automatisch beim Carriage-Return auch einen Zeilenvorschub, so sorgt der ASCII-Code 10 in Zeile 650 dafür.

EPSON-DRUCKER

Der Epson-Drucker besitzt verschiedene nationale Zeichensätze, zwischen denen wir nach Belieben mit unseren Code-Sequenzen hin und herschalten können.
Hier die Epson-Anpassung:

```
550 data 123,-1        610 data 93,-1
560 data 124,-1        620 data 27,82,2,-1
570 data 125,-1        630 data 27,82,3,35,27,
580 data 126,-1             82,2,-1
590 data 91,-1         640 data 227,-1
600 data 92,-1         650 data 27,82,2,13,10,-1
```

Die Umschaltung des nationalen Zeichensatzes erfolgt mit der Code-Sequenz 27,82. Ein weiteres Zeichen wählt den entsprechenden Zeichensatz:

27,82,0 = amerikanischer Zeichensatz
27,82,2 = deutscher Zeichensatz
27,82,3 = englischer Zeichensatz.

Wollen wir vom Drucker nicht plötzlich ein Paragraphenzeichen statt dem Klammeraffen vorgesetzt bekommen, so ist in Zeile 490 nach der 13 noch die Zahl 64 einzufügen und in Zeile 651 die Code-Sequenz:

```
651 data 17,82,0,64,27,82,2,-1
```

Leider mißfällt uns, daß der Epson-Drucker kein „π" in seinem Zeichensatz besitzt. Wir kamen noch nicht dazu, ein solches durch Drucker-Programmierung oder Einzelnadelansteuerung zu realisieren, und ließen verschämt den IBM-Code in Zeile 640 einfach stehen, werden dieses aber zu gegebener Zeit nachholen.

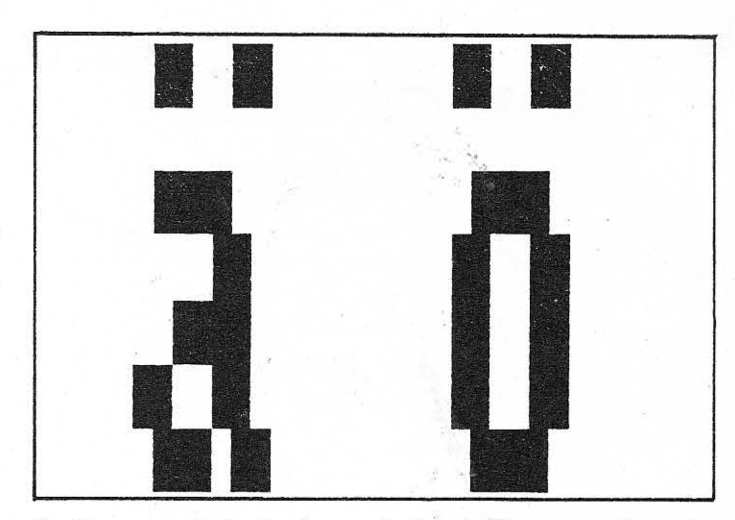
DRUCKER OHNE DEUTSCHEN ZEICHENSATZ

Billigen Matrix-Druckern geht leider oft ein deutscher Zeichensatz ab. Dies zu beheben, bereitet unserer DRUCKER-ANPASSUNG aber keine allzu großen Schwierigkeiten. Für Commodore-Drucker, ausgenommen die Farbdruckern MCS 801, verwenden wir folgende Codes:

```
550 data 8,2,160,208,133,208,132,208,133,184,
    192,128,128,128,15,-1
560 data 8,2,128,184,197,128,196,128,197,184,
    128,128,128,128,15,-1
570 data 8,2,128,188,129,192,128,192,129,188,
    192,128,128,128,15,-1
580 data 8,2,128,248,132,130,129,136,193,150,
    192,160,128,128,15,-1
590 data 8,2,224,145,168,132,162,132,168,145,
    224,128,128,128,15,-1
600 data 8,2,184,197,128,196,128,196,128,197,
    184,128,128,128,15,-1
610 data 8,2,188,193,128,192,128,192,128,193,
    188,128,128,128,15,-1
620 data 17,-1
630 data 92,-1
640 data 8,2,128,136,128,252,128,132,128,193,
    188,128,128,128,15,-1
650 data 13,-1
```

DEUTSCHE ZEICHEN MIT DEM MCS-801

Leider unterscheiden sich sowohl der Betrieb dieses Druckers im Grafikmodus als auch das Aussehen der Zeichen sich von anderen Commodore-Druckern. Die Zeichen haben eine 8*8-Matrix. Im Handbuch ist das Zeichenmuster gut ablesbar. Was fehlt, sind deutsche Zeichen, die gut zu den dort sich befindenden passen. Wir kamen noch nicht dazu, diese zu kreieren, werden dies aber noch nachtragen. Vorerst sagen wir Ihnen, wie Sie selbst eigene Zeichen entwerfen können. Mit der Code-Sequenz 27,75 schalten wir in den Grafikmodus. Nun hat die Anzahl der zu druckenden Bitmuster zu folgen, und zwar in Form einer dreistelligen Dezimalzahl, bei der die einzelnen Ziffern als ASCII-Codes von 48 bis 57 darzustellen sind. Der Anfang der Sequenz lautet also 27, 75, 48, 48, 56. Dann kommen die acht Bitmuster. Damit der Rechner das Sequenzende erkennt, ist die Sequenz, wie gewohnt, mit −1 abzuschließen. Nun zu den Bitmustern. Bei den Bildschirmzeichen unterteilten wir die Zeichenmatrix in acht Zeilen und stellten jede Zeile durch eine Zahl von 0 bis 255 dar. Der Drucker braucht, da seine Nadeln untereinander angebracht sind, eine Unterteilung der Matrix in acht Spalten. Für jede Spalte ist dann das Bitmuster zu berechnen. Dies geschieht nach folgendem System: Jedes Bit, das gesetzt wird, hat, je nach der Stelle, an welcher es steht, einen gewissen Wert. Von oben nach unten gelten die Werte: 1, 2, 4, 8, 16, 32, 64, 128. Befindet sich ein Punkt in der oberen Reihe der Zeichenmatrix, so zählt er also 1, befindet er sich in der unteren, so sind es 128. Die so von oben nach unten berechneten Werte der



Spalten, von links beginnend, durch Komma getrennt, sind die gesuchten Werte der an den Drucker zu sendenden Bitmuster. Wenn alles klappt, sehen wir die deutschen Zeichen dann auf dem Papier und sollten nicht vergessen, der COMMODORE-WELT-Redaktion die berechneten Werte mitzuteilen und auch eine Druckprobe, damit der beste Entwurf an die anderen MCS-801-Eigner weitergegeben werden kann, nach dem Motto, geteilte Arbeit ist halbe Arbeit.

DIN-TASTATUR AM PLUS4

Mancher unserer Leser mag bedauern, daß unser Anpassungsvorschlag leider nicht der DIN-Norm entspricht und für schreibmaschinengewohnte Tipper sich so nicht besonders eignet. Herr Burr fand nicht nur unser Zeichensatzprogramm fantastisch, sondern auch, daß damit der Verwirklichung einer DIN-Tastatur auf dem Plus4 nichts mehr im Wege steht.
Durch folgende Änderungen in SCREEN-KEYBOARD erhalten Sie eine DIN-Tastatur am Plus4:

```
315 data 59,58,64,43,93,91,186,92
320 data 169,166,90,218,89,217,164,-1
```

Script-Plus:

```
350 data 176,191,177,171,178,174,179
355 data 58,59,43,89,217,90,218,64
```

Plus4-Software:

```
350 data 251,252,253,254,219,220,221
355 data 58,59,43,89,217,90,218,64
```

Die veränderten Zeichen finden Sie dann auf folgenden Tasten:

Zeichen	**Taste**
ä	;
ö	:
ü	@
ß	+
Ä	]
Ö	[
Ü	<Shift> @
:	Ł
;	<Shift> Ł
+	<Cbm> +
y	z
Y	Z
z	y
Z	Y
<Paragraph>	<Cbm> @

Vielleicht verpassen Sie dem C16 auch eine DIN-Tastatur. Wir sind schon gespannt. *A.M.* □

PROGRAMMIERSPRACHEN ZUR AUSWAHL

SPRACHEN-GEWIRR

Der vielschichtige, schnell wachsende Markt der Programmiersprachen und ihrer Dialekte ist unüberschaubar geworden. Unsere Serie erläutert die Hauptunterschiede.

Das komplexe Thema der Programmiersprachen von Ada bis V.I.P. ist nicht nur für den Profi-Programmierer interessant, sondern auch für die Einsteiger geeignet, die sich einen Überblick der babylonischen Sprachverwirrung beschaffen wollen. Wir werden dieses Thema in mehreren Teilen aufbereiten.
Es gibt zur Zeit weltweit mehr als 160 höhere Programmiersprachen. Nur etwa 15 Prozent unterscheiden sich erheblich voneinander.
Von einigen existieren mehrere Versionen, sogenannte Dialekte. Hinzu kommen Besonderheiten der verschiedenen Computeranlagen und deren Compiler, auf denen mit den höheren Programmiersprachen gearbeitet wird. Dies fördert die Entwicklung weiterer Dialekte. Addiert man alle Variationen und Dialekte, dürfte die Anzahl der bekannt gewordenen Programmiersprachen zwischen 2000 und 2500 liegen.
Ein weiterer Faktor, der die Übersichtlichkeit auf diesem Markt einschränkt, ist die Weiterentwicklung der Sprachen. Dies bringt neben Vorteilen sicherlich auch Probleme mit sich: Bestehende Programme müssen teilweise umgestellt werden.
Hier soll nur auf die Hauptunterschiede der verschiedenen Sprachen eingegangen werden. Verbesserungen in den einzelnen Weiterentwicklungen und Dialekten ändern den Charakter der betreffenden Sprache nicht und den Anwendungsbereich nur unwesentlich. Nur dort, wo eine Sprache ihr Erscheinungsbild grundlegend geändert hat, wird im Text darauf eingegangen.
Die gebräuchlichsten Programmiersprachen werden etwas ausführlicher behandelt. Bei den einzelnen Beschreibungen werden Probleme angesprochen, die auch für andere Sprachen gelten. So greift die Besprechung über zu anderen Sprachen und allgemeinen Problemen und zeichnet Besonderheiten aus den Anwendungsgebieten auf. Zu einigen Programmiersprachen werden Beispiele gegeben; je nach Besonderheit und Bedeutung der Sprache ausführlicher dokumentiert. Sie sollen nur dazu dienen, einen optischen Eindruck zu vermitteln.

WICHTIGE ENTSCHEIDUNGSHILFE

Strukturierung und Modularität einer Sprache lassen sich so schon auf den ersten Blick erkennen. Auch verdeutlicht das Beispiel die Lesbarkeit einer Programmiersprache oder des Programms; der Charakter einer Sprache ist zu erkennen.
Die künstliche Intelligenz und Expertensysteme werden erläutert sowie Begriffe aus der Welt der Programmiersprachen. Der Leser erhält einen Eindruck, wie vielschichtig Programmiersprachen sein können. Vielleicht kann daraufhin eine Kauf- oder Lern-Entscheidung einfacher gestaltet werden. Zumindest wird der Leser sich ein eigenes Urteil bilden können, wenn die eine oder andere Programmiersprache angepriesen oder verurteilt wird. In den meisten Fällen werden hier nur Vor- und Nachteile einer Programmiersprache genannt und das vorrangige Einsatzgebiet. Es gibt keine „schlechten" Programmiersprachen, sie müssen immer in ihrem Umfeld betrachtet werden, und sei es nur das zeitliche. Auf veraltete Sprachen wird gesondert hingewiesen.
Die vorliegende Auswahl von Programmiersprachen ist nicht vollständig. Sie kann dies zu keinem Zeitpunkt sein, da in immer kürzeren Abständen neue Dialekte und „Verbesserungen" auf den Markt geworfen werden. Es erschien auch nicht sinnvoll, hier nur auf genormte Sprachen einzugehen, da sie zum einen rar sind, zum anderen keinen Überblick über die momentane Marktsituation geben können, denn Normierungen sind langwierige Unterfangen.
Andererseits kann unmöglich jeder firmeneigene Sprach-Dialekt ausführlich besprochen werden, daher sind sie zum Teil nur der Übersicht wegen namentlich aufgelistet oder wegen ihrer Sonderstellung kurz angesprochen.
Auch sprachliche Programmiermethoden sind in diese Liste aufgenommen. Die Grenze zwischen Methode und Sprache ist oft fließend. Daher erscheint manchmal der Begriff Programmiersystem anstelle der Programmiersprache. Dies sollte nicht verwirren, denn auch eine Programmiersprache ist immer in einem System zu sehen, und sei es nur der spracheigene Compiler oder Interpreter.

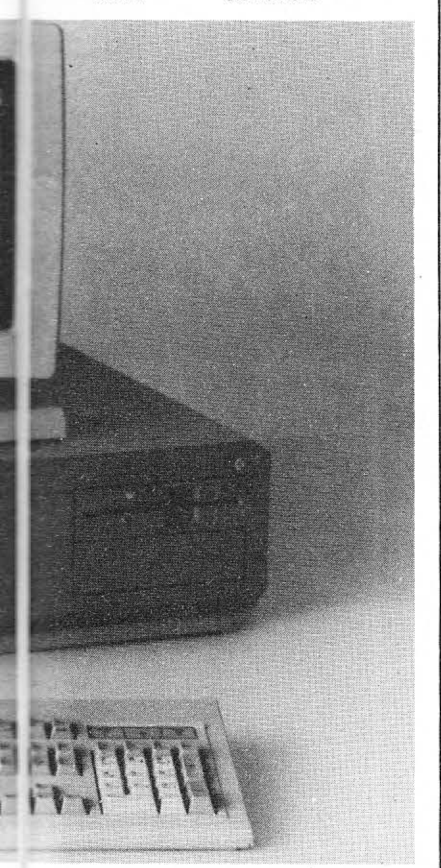


Moderne Computersysteme verlangen moderne Programmiersprachen

1. Ada

Ada ist wie HOL (siehe dort) eine Systemimplementierungssprache, die in den USA vom US-Verteidigungsministerium für den militärischen Einsatz entwickelt wurde. Seit dem 1. Januar 1985 ist Ada für alle US-Waffen-

systeme obligatorische Programmiersprache und löste damit etwa 800 (achthundert!) Programmiersprachen aus diesem Anwendungsbereich ab. Da Ada aus dem militärischen Bereich kommt, hatte sie ursprünglich einen Decknamen: green.
Ende der 70er Jahre hatte das DoD (US Departement of Defense) die Entwicklung einer Programmiersprache ausgeschrieben. Es wurden insgesamt 17 Vorschläge eingereicht, von denen vier in die engere Wahl gezogen wurden. Diesen ordnete man Tarnnamen zu:
green – Honeywell Bull
red – Intermetrics
blue – Softech



yellow – SRI International.

Das DoD entschied sich für „green" und so bekam Honeywell Bull den Zuschlag.
Ada ist eine Real-Time-Sprache (= Echtzeitsprache) ähnlich Pearl, jedoch unterscheiden sich beide unter anderem in dem aufzubringenden Programmieraufwand stark.
Ada wurde ursprünglich entwickelt für den Einsatz in embedded systems, das sind integrierte Computersysteme, die innerhalb größerer Systeme „eingebettet" sind und dort bestimmte Aufgaben übernehmen.
Beispiele für embedded systems sind unter anderem: Flugüberwachungssysteme, Feuerleitsysteme usw., aber auch mikroprozessorgesteuerte Waschmaschinen. Die Entwicklung von Programmen für solche Systeme ist in der Regel wesentlich kostenintensiver als technisch-wissenschaftliche oder kommerzielle Programme, da die letztgenannten Bereiche mit standardisierten Sprachen abgedeckt sind (Fortran, Cobol . . .).
Ada hat sich zwischenzeitlich auch auf dem nichtmilitärischen Bereich etabliert.

Programmiersprachen

1 Ada
2 ADAPT
3 ADF
4 Algol
5 APL
6 APS
7 APT
8 Assembler
9 ATLAS
10 Autocode
11 Autopit
12 Autopol
13 Autoproc
14 Autopromt
15 Autospot
16 BASIC
17 BCPL
18 BLISS-32
19 C
20 CDL/2
21 Cobol
22 Comal 80
23 Coursewriter
24 CSL
25 DIPOL
26 DL
27 Elan
28 EOL
29 Exapt
30 FOCUS
31 Forth
32 Fortran
33 GPL
34 HOL
35 HOPE
36 Lisp
37 Logo
38 LPG
39 MACRO 80
40 MANTIS
41 Modula 2
42 mu-SIMP
43 Natural
44 NEAT
45 NELIAC
46 NPL
47 NUCOL
48 OPS-5
49 OCCAM
50 Pascal
51 Pearl
52 PILOT
53 PLANKALKÜL
54 PL/1
55 PROLOG
56 PROSA
57 QBE
58 Query Language(QL)
59 PRG
60 RPNL
61 SIMULA
62 SL 3
63 Smalltalk
64 SQL
65 SYMAP
66 V.I.P
67 Verschiedene

ADA: WECHSEL INS ZIVILLEBEN

Obwohl Ada eine höhere Programmiersprache, also maschinenunabhängig ist, enthält ihr Sprachumfang maschinennahe Elemente, um dem Benutzer die Möglichkeit zu geben, für die – im militärischen Bereich besonders aufwendigen – Spezialanwendungen Ein- und Ausgabefunktionen anwendungsspezifisch zu erstellen.
Ada gibt es seit 1981 und hat in außermilitärischen Anwendungsgebieten Fuß gefaßt; sowohl im technisch-wissenschaftlichen Bereich, als auch in der kommerziell-administrativen Datenverarbeitung.
Ada ist inzwischen eine leistungsfähige, universelle Programmiersprache mit integrierten Spracheinrichtungen, die eine Vielzahl von modernen Programmiertechniken unterstützt. Ada wurde unter dem Aspekt der Wartungsfreundlichkeit sowie der gesteigerten Zuverlässigkeit und Übertragbarkeit entwickelt. Die Vorteile kommen besonders zur Geltung bei rechenintensiven Programmen in Echtzeitumgebung.
Ada – vorwiegend auf größeren Rechenanlagen einsetzbar – verlangt eine strenge Zuweisung von Datentypen in der ersten Phase der Programmentwicklung. Da die Überprüfung der Datentypenzuweisungen schon während der Compilierung stattfindet, können folgenschwere Fehler vermieden werden.
Die Datenabstraktion ermöglicht es dem Benutzer, sich auf die wesentlichen Punkte seines Programmes zu konzentrieren, ohne sich um die darunterliegenden Einzelheiten kümmern zu müssen. Die Details der Implementierung bleiben für ihn verborgen, während ihm gleichzeitig alle Mechanismen für ihre Verwendung zur Verfügung stehen.
Auffallend ist hier die Vielzahl der zur Verfügung stehenden Datentypen im Gegensatz zu anderen Programmiersprachen.
Die *Datentypen* gliedern sich in Ada wie folgt auf:

1. Ada Datentypen
1.1. Verweistypen
1.2. strukturierte Datentypen
 - Ableitungstypen
 - Unterbereichstypen
 - abstrakte Typen
1.2.1. Felder (Strings)
1.2.2. Verbunde
1.2.2.1. mit Varianten
1.2.2.2. ohne Varianten
1.3. skalare Datentypen
 - Ableitungstypen
 - Unterbereichstypen
 - abstrakte Typen
1.3.1. diskrete Datentypen (indiex-/iterationsfähig)
1.3.1.1. ganzzahlige Datentypen
1.3.1.2. Aufzählungstypen (z.B. Boolean, Character)
1.3.2. reelle Datentypen
1.3.2.1. Festpunkttypen (mit absoluter Genauigkeit)
1.3.2.2. Gleitpunkttypen (mit relativer Genauigkeit).

Ada bietet hervorragende Sprachelemente zur Steuerung paralleler Programmabläufe, ohne daß auf die Systemdienste des Betriebssystems zurückgegriffen werden muß. Der Programmierer hat die Möglichkeit, große Programme zu unterteilen und getrennt zu compilieren.
Dabei liefert die Bibliotheksverwaltung Informationen zu den übrigen Compiler-Einheiten. Andere Programmiersprachen stellen nur wenig Information bei getrennt compilierten, aber voneinander abhängigen Programm-Modulen bereit.
Eine besondere Stärke von Ada sind die generischen Definitionen, spezielle Formen von Programmeinheiten, die Algorithmen definieren, analog zur Definition von Unterprogrammen, wobei die Datentypen variabel bleiben und erst bei der Compilierung festgelegt werden.

EINFACHE FEHLERBEHANDLUNG

Die Fehlerbehandlung bei Ada ist vorbildlich und trägt entscheidend zur Wartungsfreundlichkeit bei.
Die Namensgebung von Ada hat nichts mit einer Abkürzung zu tun, wie dies bei vielen anderen Programmiersprachen der Fall ist, sondern hat folgenden Ursprung: Der britische Mathematiker Charles Babbage (1792 bis 1871) entwickelte ein Gedankenmodell einer Analytical Engine, eines Computers. Dieses Modell verfügte über eine heute noch aktuelle Architektur: Eingabe, Ausgabe, Verarbeitung, Speicher. Seine Maschine konnte niemals gebaut werden, hätte doch alles mechanisch bewerkstelligt werden müssen, mit unendlich vielen Zahnrädern und ähnlichem. Babbage wurde nicht für ernst genommen. Moralisch und finanziell wurde er unterstützt von einer Edeldame namens Ada Augusta Countess of Lovelace. Ihr wurde diese Programmiersprache gewidmet.
Ada unterstützt sowohl die Top-Down- als auch die Bottom-Up-Entwicklung; beide Methoden sind gleichermaßen gebräuchlich, wenn sie auch mit sehr unterschiedlichen Vor- und Nachteilen behaftet sind.
Beide Verfahren beschreiben die Vorgehensweise bei der Programmerstellung im allgemeinen – also nicht nur bei Ada, sondern bei Programmiersprachen überhaupt.

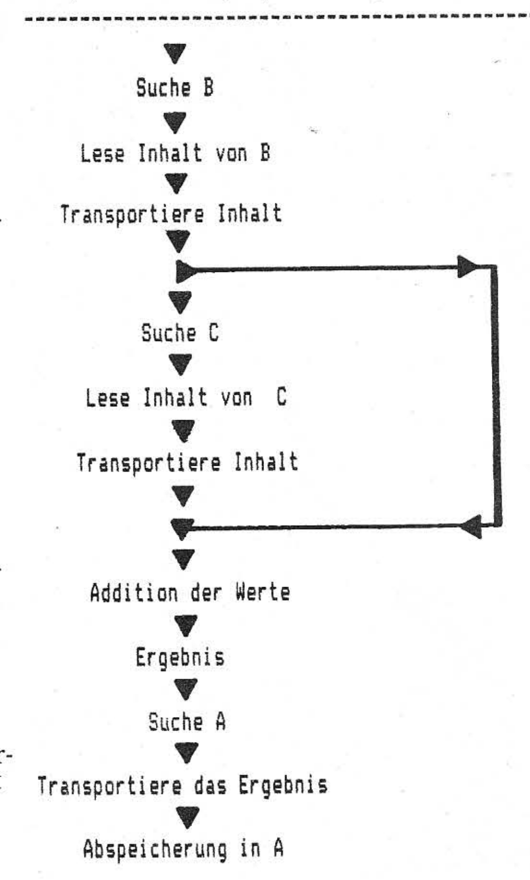
1. Top-Down-Methode
Das zu erstellende Programm wird durch schrittweise Verfeinerung entwickelt. In der Praxis sieht das so aus, daß die Problemlösung – der Weg – grob vorgezeichnet wird, so daß ein Programmgerüst entsteht. Verschiedene diffizile und oft benötigte Randprobleme, wie Datei-Ein- und -Ausgabe, werden dann in Form von Unterprogrammen realisiert. Ada unterstützt die Top-Down-Methode unter anderem durch die Möglichkeit der Einflechtung von body stubs anstelle der Unterprogramme. Body stubs sind Platzhalter für den Korpus (Rumpf) eines Unterprogrammes. Die eigentlichen Unterprogramme können später erstellt und sogar separat compiliert werden.

2. Bottom-Up-Methode
Diese Vorgehensweise bei der Programmierung steht in krassem Gegensatz zur Top-Down-Entwicklung. Hier wird das Problem zunächst untergliedert, und Einzellösungen in Form von Unterprogrammen werden ausgearbeitet. Daraufhin kann der Rumpf des Programms mit den entsprechenden Unterprogrammaufrufen erstellt werden.

Der große Vorteil von

Tabelle 1:

Beispiel Assemblerprogrammierung Ablaufplan:



Ada ist die strenge Standardisierung. Darüber wacht das Ada Joint Programm Office, dem sogar Lehrbücher über Ada vorgelegt werden müssen.
Ada ist als Universalsprache zu bezeichnen, die dem professionellen Softwareentwickler eine umfassende Sprachunterstützung bietet.
Der Programmieranfänger wird sich mit Ada jedoch nicht leicht tun, das ist eben der Preis der Komplexität einer höheren Programmiersprache.
Ada-Experten gehen sogar so weit, zu behaupten, daß man Fähigkeiten dieser mächtigen Sprache nur mit einem abgeschlossenen Informatikstudium ausnutzen könne. Mit anderen Worten: Ada ist nicht geeignet für Anfänger.

2. ADAPT
NC(Numerical Controlled)-Steuersprache (siehe APT, EXAPT).

3. ADF
Dedizierte Abfrage-Sprache im Bereich informativer Systeme (Datenbanken etc.), siehe hierzu auch Query Language oder SQL.

4. Algol
Algorithmic **L**anguage.
Algorithmische Sprache.
Eine mathematisch-wissenschaftlich orientierte Sprache, die in der Hauptsache an Hochschulen angewendet wird.
Algol ist seit 1959 auf dem Markt und hat starke Ähnlichkeiten mit Fortran, ist jedoch systematischer in ihrem Aufbau des Sprachschatzes und umfangreicher. Die Ausgabe von 1959 wurde bereits 1960 revidiert (Algol 60). 1968 erschien dann Algol 68, in ihrem Wortschatz wesentlich umfassender und ausgereifter. Wie bei vielen anderen Programmiersprachen sind auch hier die

Tabelle 2:

```
--------------------------------------------------------------------------
Programmbeispiel in Assembler für CPU 6502 von MOS-Technology,
zu finden im VC20/C64/Apple etc.
Source-Listing Ausschnitt zu Assembler:
--------------------------------------------------------------------------
1A24          91 26      STAIZ TEMPL    Abspeichern und nächste
1A26  GENADR  20 BD 19   JSR NXTADR     Adresse holen, wenn 0,
1A29          F0 D8      BEQ AGAIN      dann nicht fertig, sonst
1A2B          20 A6 19   JSR RSTORE     Rücksprung zu RSTORE
1A2E          69         RTS
1A2F  NBRS    98         TYA            X und Y in Stack speichern
1A30          48         PHA
1A31          8A         TXA
1A32          48         PHA
1A33          A0 00      LDYIM $00      Y und N gleich Null setzen
1A35          84 32      STYZ
1A37          A2 08      LDYIM $08      Acht Nachbarn testen
1A39  OFFS    B5 29      LDAZX OFFSET
1A3B          10 15      BPL ADD        Springe nach ADD, wenn
1A3D          49 FF      EORIM $FF      OFFSET positiv
1A3F          85 37      STAZ TMP       sonst Subtraktion vorbereiten,
1A41          38         SEC            Carry-Bit setzen
1A42          A5 39      LDAZ RCSH
1A44          E5 37      SBCZ TMP       Subtrahieren, damit die
1A46          85 22      STAZ CHL       korrekte Adresse des
1A48          A5 3A      LDAZ RCSH      Nachbarn entsteht,
1A4A          85 23      STAZ CHH       Sprung zur
1A4C          B0 11      BCS EXAM       Untersuchungsroutine
1A4E          C6 23      DECZ CHH       Page-Grenze überschreiten
1A50          D0 0D      BNE EXAM       weiter zu EXAM
1A52  ADD     18         CLC            Addieren vorbereiten, wenn
1A53          65 39      ADCZ RCSL      OFFSET positiv, Ergebnis
1A55          85 22      STAZ CHL       in CHL ablegen.
1A57          A5 3A      LDAZ RCSH
1A59          85 23      STAZ CHH
1A5B          90 02      BCC EXAM       Sprung zur EXAM-Routine
1A5D          E6 23      INCZ N         Page-Grenze überschreiten
1A5E  EXAM    B1 22      LDAIY CHL      Nachbardatenbyte holen
1A61          C9 51      CMPIM DOT      und prüfen, ob besetzt
1A63          D0 02      BNE NEXT
1A65          E6 32      INCZ N         Anzahl der Nachbarn
1A67  NEXT    CA         DEX            erhöhen
1A68          D0 0F      BNE OFFS       Sprung nach OFFS
1A6A          68         PLA            X und Y Werte aus Stack
1A6B          AA         TAX            zurückholen
1A6C          68         PLA
1A6D          A8         TAY
1A6E          60         RTS
--------------------------------------------------------------------------
```

verschiedenen Ausgaben kaum oder nur bedingt kompatibel.
Diese Sprache weist starke Nachteile in der Ein- und Ausgabe von Daten auf. Ihre Vorteile liegen in der streng mathematischen Ausdrucksform.
Algol ist für Deutschland in der DIN 66026 „Informationsverarbeitung; Programmiersprache Algol" festgelegt.
Nach Algol 60 und Algol 68 folgte noch ein Versuch, diese „veraltete" Programmiersprache zu retten: Algol W, aber ohne nennenswerten Erfolg.
Pascal war der nächste Entwicklungsschritt und löste damit die Algol-Familie ab.

5. APL

A Programming Language.
APL ist ähnlich wie BASIC eine problemorientierte Sprache. Sie ist 1960 in den USA von dem Mathematiker K. Iverson entwickelt worden und wurde in erster Linie bei dem Time Sharing und bei größeren Rechenanlagen eingesetzt. 1962 wurde APL publiziert, aber erst 1966 gab es die erste Implementation für das IBM-System /360.
Mittlerweile gibt es auch schon mehrere Interpreter für diverse Microcomputer.
Time Sharing (Zeit-Aufteilung) einer Rechenanlage liegt dann vor, wenn mehrere Benutzer zur selben Zeit an einem Rechner angeschlossen sind. Der Rechner kann aber immer nur ein Programm bearbeiten, daher muß die Benutzerzeit aufgeteilt werden.
Time Sharing ist das Zuweisen von Arbeitszeit (meist im Sekunden-Bereich) der Zentraleinheit des Rechners an die verschiedenen Benutzer; es ist jedoch auch mit anderen Programmiersprachen möglich. APL spielt heute im Zeitalter der Microcomputer keine große Rolle mehr in bezug auf das Time Sharing.
Das Hauptanwendungsgebiet für APL liegt in der Wirtschaftsmathematik und Statistik sowie in der Mathematik bei der Vektor- und Matrizenrechnung. Vorteile gegenüber Fortran und Algol: bessere Datenhandhabung und bessere Dialogmöglichkeiten. Im Gegensatz zu diesen ist APL jedoch interpretativ: Jeder Befehl wird sofort interpretiert und in eine Maschinensprache umgeformt (kein Compiler notwendig). Außerdem ist APL fünf- bis achtmal schneller als Cobol und Fortran.

Weitere Vorteile von APL:

- APL bietet sowohl gute Berechnungsmöglichkeiten als auch eine gute Textverarbeitung;
- Durch wenige Reglements und wenige Symbole ist APL sehr kompakt und leicht erlernbar für den Anfänger;
- Die zu bearbeitenden Daten müssen nicht wie in anderen Sprachen vom Benutzer genau definiert werden. Es muß keine Rücksicht auf vorgegebene Formate und Größenordnungen der Daten genommen werden;
- Problemlose Matritzen- und Vektorverarbeitung sowie einfache Programmierung von Tabellenausgaben etc.

APL unterscheidet sich in einigen Punkten sehr stark von anderen Programmiersprachen, und obwohl schon etwas betagt, gibt es keine vergleichbare Sprache.
Sie ist nicht nur eine höhere Programmiersprache zur Beschreibung von Algorithmen (problemorientiert), sondern auch eine moderne Notation für die Darstellung mathematischer Sachverhalte. Komplizierte mathematische Zusammenhänge können in APL vereinfacht dargestellt werden. In APL ist der Umgang mit komplexen Objekten wie Vektoren, Matritzen und Folgen genauso einfach wie mit einfachen Zahlen. Komplizierte mathematische Zusammenhänge können so einfach und verständlich dargestellt werden.

In APL gibt es drei Gruppen von *Operatoren:*

1. Skalare Operatoren. Diese sind für skalare Argumente definiert, die als Ergebnisse wiederum Skalare liefern. Die skalaren Operatoren sind jedoch auch auf beliebige Strukturgrößen anwendbar, allerdings muß bei dyadischen Funktionen entweder die Struktur von Links- und Rechtsargument identisch oder eines der beiden Argumente ein Skalar sein.
2. Gemischte Operatoren. Hier treten stets Vektoren als Argumente oder als Ergebnis auf. Sie lassen sich daher nicht ohne weiteres auf Argumente höheren Ranges erweitern.
3. Spezielle Operatoren. Alle übrigen Operatoren, die nicht zu 1. oder 2. gehören, werden als spezielle Operatoren bezeichnet.

Das Hauptmerkmal von APL liegt darin, daß ein

in der Notation von Iverson dargestelltes Problem bereits das Programm zu dessen Lösung darstellt. Zur Programmgestaltung stehen ergänzend zur Verfügung: die Möglichkeit zur Programmstrukturierung (Verzweigung, Schleifen, Unterprogrammbildung) und die Namensgebung für das Programm. Nicht zuletzt zu nennen ist die Möglichkeit, Programme zu editieren, korrigieren, abzuspeichern etc.; aber das zeichnet ja alle Programmiersprachen aus.
Oft ist es nicht erforderlich, ein Problem als APL-Programm zu bearbeiten. Vielmehr wird das Problem mit mehreren Anweisungen Schritt für Schritt der Lösung zugeführt. Sobald ein Problem mehrfach auftritt – nur mit wechselnden Parametern – wird eine Funktion auch programmiert.
APL ist ein interpretatives System. Jede eingegebene Anweisung wird sofort in eine systeminterne tabellarische Form umgewandelt und ausgeführt. Einzige Ausnahme sind Programmzeilen mit definierten Funktionen. Diese werden zum späteren Gebrauch abgespeichert.
Wie bei allen interpretativen Programmiersprachen erfolgt hier eine direkte Fehlererkennung und -meldung, was die Programmierung erleichtert.

DESKTOP FÜR PROGRAMMIERER

Die Arbeit in APL ist vergleichbar mit der Arbeit am Schreibtisch (nicht zu verwechseln mit Desk Top Publishing): Man macht sich Notizen, führt Berechnungen durch, hält Zwischenergebnisse fest, faßt Ergebnisfolgen in Tabellen zusammen und räumt von Zeit zu Zeit den Schreibtisch wieder auf, indem man nicht mehr benötigte Notizen in den Papierkorb wirft.

Tabelle 3:

```
--------------------------------------------------------------
Programmbeispiel zu Autopit:
Listing
--------------------------------------------------------------
PARTNO/4848 FLANSCH
MACHIN/1
REMARK/BESCHREIBUNG DES ROHLINGS
CONTUR/BLANK
BEGIN/-3,40,YLARGE,PLAN,-3
RGT/DIA,150
RGT/PLAN,57
RGT/DIA,80
TERMCO
REMARK/BESCHREIBUNG DES ROHLINGS
CONTUR/PART
SURFIN/FIN
M10,N10,BEGINN/13,77.5,XSMALL,DIA,85,BEVEL,1,FIT,3,OSETNO,2
M0,RGT/PLAN,0
L2=LINE/0,68.9,ATANGL,60,
M1,RGT/L2
M2,RDT/DIA,146,RGH,FIT,4,OSTEND,1
L4=LINE/0,68.5,80,ATANGL,-75
M3,RGT/L4,FEED,0.8
M4,N5,RGT/PLAN,26,ROUND,5
M6,N7,LFT/DIA,120,BEVEL,3,FIT,3
M8,N9,RGT/PLAN,56,BEVEL,1
L9=LINE/53.43.1,15,42.5
RGT/L9
TERMCO
REMARK/TECHNOLOGISCHE DEFINITIONEN
PART/CTR,250,TORLIM,100
CLAMP/NORMAL,DRL,3,ZPS,15
CHUCK/1,DCL,40,MHJ,60,DJ,300
PPRINT/WERKSTUECK AM AUSSENDURCHMESSER EINSPANNEN
REMARK/BEARBEITUNGSDEFINATIONEN
A1=FACE/RGH,TOOL,112,9,SPEED,145
A2=TURN/RGH,TOOL,113,9,SPEED,145
A4=CONT/RGH,TOOL,114,9,SPEED,145
A5=TURN/FIN,TOOL,212,11,SPEED,120
A6=CONT/FIN,TOOL,214,11,SPEED,80
REMARK/BEARBEITUNG DES WERKSTUECKES
CUT/A1,M8,TO,N9
CUT/A4,N9,TO,N10
CUT/A2,M8,RE,M3
CUT/A4,M8,RE,M3
CUT/A6,N9,RE,M3,FEED,.01
CUT/A6,N9,TO,M10,FEED,.05
CLAMP/INVERS;DRL;56,ZPS,20
PPRINT/WERKSTUECK UMSPANNEN
B1=THREAD/TOOL,217,1,SPEED,120
CUT/A2,M2,TO,M3
CUT/A1,M1,RE,M0
CUT/A6,M0,TO,M3,FEED,.05
CUT/A5,M10,RE,M10,FEED,.02
CUT/B1,M2,TO,M3,DEPTH,2.0,DIRECT,25,PITCH,2,1,PASSES,6,1
FINI/
--------------------------------------------------------------
```

Dieser Arbeitsumgebung entspricht in APL der aktive Workspace (Arbeitsbereich).
Jedem APL-Benutzer steht ein solcher Workspace mit jeweils 32 K-Byte Speicherplatz zur Verfügung. Hierin finden alle Eintragungen, Programmabläufe, Löschungen usw. statt. Werden Programme gestartet oder soll auf Variablen zugegriffen werden, so müssen diese im aktiven Workspace vorhanden sein. Verschiedene Workspaces werden in Bibliotheken gespeichert. Jedem Benutzer ist eine Bibliothek

DATENSCHUTZ BLEIBT GEWAHRT

zugeordnet, mit der Möglichkeit einer Schutzkennung. Die hier abgelegten Workspaces werden als passiv bezeichnet. Aktiv ist immer nur eines – das aktuelle Workspace. Dieses befindet sich im Arbeitsspeicher des Rechners (bei mehreren Anwendern auch mehrere Workspaces), während sich die passiven Workspaces immer auf externen Datenträgern befinden (Festplatte o.ä.). Workspaces können vor dem Zugriff anderer Benutzer geschützt werden. Auf ungeschützte Workspaces können alle Benutzer zugreifen, allerdings nur lesend, nie schreibend.
Der Inhalt des aktiven Workspace kann mit Hilfe von Systemanweisungen geändert werden. Außerdem existieren Anweisungen für die Manipulation der Workspace-Bibliotheken und für die Datenübertragung zwischen verschiedenen Benutzern.
Alle Systemanweisungen beginnen mit einer Klammer ")", nach links offen. Zwischen den Systemanweisungen und den APL-Ausdrücken muß streng unterschieden werden; so dürfen diese beiden Befehlstypen nicht gemischt werden.
Jedem Workspace muß

Tabelle 4:

```
-----------------------------------------------------------------------------
Programmbeispiel in BASIC
in Standard-BASIC
Listing-Ausschnitt
-----------------------------------------------------------------------------
5 REM ********************************************* DIMENSIONIERUNG 100 PROBEN *
10 DIM P(100)
20 GOSUB 500
25 REM *************************************************** BILDSCHIRM LOESCHEN *
30 PRINTCHR$(147)
40 PRINT"WIEVIELE PROBEN SOLLEN UNTERSUCHT WERDEN"
45 PRINT
50 INPUT N
60 FOR I = 1 TO N
70 IF I = 9 THEN 90
80 PRINT I:PRINT TAB(2) "PROBE":GOTO 100
90 PRINT I:PRINT TAB(3) "PROBE"
100 INPUT P(I)
110 NEXT I
120 PRINT "ANZAHL DER PROBEN   =" N
130 REM **************************************************** ORDNEN DER PROBEN *
140 LET L = N - 1
150 FOR J = 1 TO L
160 IF P(J) = P(K) THEN GOTO 190
170 LET C = P(J)
180 LET P(J) = P(K)
190 NEXT K
200 NEXT J
210 PRINT "KLEINSTER WERT      =" P(N)
220 PRINT "GROESSTER WERT      =" P(I)
230 REM ********************************************************* MITTELWERT *
240 LET M = 0
250 FOR I = 1 TO N
260 LET M = M + P(I)
270 NEXT I
280 LET M = M/N
290 PRINT "MITTELWERT          =" M
300 LET X = 0
310 FOR I = 1 TO N
320 LET X = X + (P(I) - M).2
330 NEXT I
340 REM **************************************** VARIANZ UND STANDARTABWEICHUNG *
350 LET S2 = 1 / (N - 1) * X
360 LET S = SQR (S2)
370 PRINT "VARIANZ             =" S2
380 PRINT "STANDARTABWEICHUNG  =" S
390 REM *************************************************** VARIANZKOEFFIZIENT *
400 LET V = (S / M) * 100
410 PRINT "VARIANZKOEFFIZENT   =" V "%"
420 GET R$ : IF R$ = "" THEN GOTO 420
430 REM ***************************** AUSDRUCK DER PROBEN IN GEORDNETER REIHE *
440 FOR I = 1 TO N
450 PRINT P(I)
460 NEXT I
500 PRINT "TEST"
510 RETURN
-----------------------------------------------------------------------------
```

ein Name gegeben werden, unter dem es aufgerufen und bearbeitet oder aber abgespeichert und in der Bibliothek wieder aufgefunden werden kann. Der Name des Workspace kann jeder beliebige in APL zugelassene sein, allerdings sind nur die ersten elf Zeichen relevant für die Identifikation. Der Workspace in APL besteht in der Regel aus Programmen und Daten (in Form von Variablen). Jede Funktion und jede Variable erhält einen eindeutigen Namen. Die Namensgebung unterliegt strengen Definitionsregeln. Zusätzlich ist im Workspace ein Bereich für Symboltabellen reserviert, in dem alle Namen der vorhandenen Objekte eingetragen werden. In der Regel sind dies nicht mehr als 256 Eintragungen. Diese Voreinstellung kann jedoch geändert werden.

TASTATURBELEGUNG NOTWENDIG

Namen können zu Gruppen zusammengefaßt werden, welche ihrerseits wieder mit Namen gekennzeichnet werden. Auch hier sind genaue Regeln zu beachten. Namen in diesen Listen müssen nicht zwangsläufig auch belegt sein. Diese Namensliste ist manipulierbar mit verschiedenen Befehlen. So kann man sich zum Beispiel eine Liste alphabetisch ordnen lassen, getrennt nach Variablen, Funktionen oder Gruppen.
APL benötigt eine besondere Tastatur oder zumindest eine Neubelegung der Tasten, denn APL arbeitet mit einer ganzen Reihe von Sonderzeichen. Diese sind in erster Linie Definitionen von APL-Funktionen (wie zum Beispiel das Wurzelzeichen in der Mathematik).
APL kennt keine speziellen Ein- und Ausgabeanweisungen. Daten werden zusammen mit den Operatoren eingegeben. Alphanumerische Daten

werden in Hochkommas gesetzt. Eingaben werden stets um sechs Zeichen nach rechts eingerückt eingegeben. Ausgaben erfolgen in der ersten Spalte beginnend. Verschiedene Rechnerimplementationen unterstützen die Hard- und Softwareeigenschaften spezieller Computer. APL-Plus für den Apple-Macintosh beispielsweise verwaltet Pull-Down-Menüs, Ikondarstellung, grafische Eigenschaften sowie Mauseingabe.

6. APS
Assembly Programming System.
Maschinenorientierte Programmiersprache.
APS ist eine spezielle Ausführung von Assembler (siehe dort).

7. APT
Automatically Programmed Tools.
Anwendung nur im Maschinenbau zum numerischen Steuern von Werkzeugmaschinen. Es gibt viele Versionen von APT, die alle in erster Linie von den zu steuernden Maschinen abhängen.
Beispiel: die Teilsprache EXAPT. Sie wird für Maschinen eingesetzt, die bohren, fräsen und drehen.
APT und EXAPT sind die wichtigsten Sprachen zur Steuerung von NC-Maschinen.
APT ist besonders für Mehrachsprobleme geeignet.
Die gewünschte Bahnkurve wird in Polygonpunkte zerlegt, die geradlinig verbunden werden. Die räumliche Kurve wird somit durch lineare Interpolation erzeugt.
APT wurde 1963 in den USA erstellt und zwei Jahre später in Deutschland zu EXAPT weiterentwickelt.
Seit 1967 besteht in Deutschland ein „APT und EXAPT-Verein" in Aachen.

8. Assembler
Maschinenorientierte Programmiersprache.

Tabelle 5:

```
-------------------------------------------------------------
Programmbeispiel zu C
Strukturiertes Listing
-------------------------------------------------------------
/*      Primzahlenprogramm "Sieb des Erathostenes"
        (nach BYTE 1/1983, mit Korrekturen)
*/

$include (stdio.h)

$define TRUE           1
$define FALSE          0
$define DURCHL         250
$define MENGE          8190

char flags(MENGE +1) = (0);

main () (

     register int i, k;
     static int lauf, zaehler;

     printf("%d Durchlaeufe: ", DURCHL);
     for(lauf = 1;lauf(= DURCHL; lauf++) (
             zaehler = 0;
             for(i = 0; i = (= MENGE; i++)
                    flags(i) = TRUE;
             for(i = 2; i (= MENGE; i++) (
                    if(flags(i)) (
                         for(k = i + i; k (= MENGE; k += i)
                              flags(k) = FALSE;
                         zaehler++;

                         )
                    )
             )
     printf("%d Primzahlen/n" , zaehler);
     exit (0);
     )
-------------------------------------------------------------
```

Assembler-Befehle sind dem eigentlichen Maschinencode sehr ähnlich: Jeder Befehl, jedes Sprachelement entspricht genau einem Maschinenbefehl.
Die Struktur der einzelnen Befehle ist primitiv einfach: Befehlscode, Adresse 1, Adresse 2, Adresse 3. Es gibt Ein-, Zwei- und Drei-Adreß-Assembler, je nachdem, wie viele Adressen hinter dem eigentlichen Befehl stehen können.
Liegt ein sogenanntes Ein-Adreß-System vor, wird ein Akkumulator (zentrales Register) benötigt. Jeder Befehl steuert den Akkumulator an und verknüpft dessen Inhalt mit der im Befehl angegebenen Adresse.
So müssen selbst einfache Anweisungen bei der Programmierung schrittweise zergliedert werden (siehe Tabelle 1).
Zwei- und Drei-Adreß-Assembler sind entsprechend kompakter und schneller in der Ausführung, setzen aber einen anderen Maschinentyp voraus.
Für den Befehlscode sind Kürzel eingeführt worden, die sogenannten mnemotechnischen Abkürzungen (Mnemotechnik = Ersetzen von Zeichenketten durch Worte und Kürzel, die Bezug zur Bedeutung der Zeichenkette haben).
Programme müssen mit einem Assembler-Programm (ähnlich einem Compiler oder Interpreter bei anderen Sprachen) übersetzt werden, was allerdings durch die Kürze und Einfachheit der Befehle sehr rasch geschieht.
Das Assembler-Programm liefert zu dem vom Programmierer geschriebenen Quellprogramm ein Objektprogramm. Für den umgekehrten Fall kann man einen Disassembler verwenden, der den Maschinencode (Objektprogramm) wieder in lesbare mnemotechnische Befehlsfolgen umwandelt (Quelle).

ASSEMBLER BRINGT GESCHWINDIGKEIT

Hierin sind auch die Vorteile von Assembler zu sehen: die Ausführungsgeschwindigkeit der Programme und der geringe Speicherplatzbedarf. Beides Vorteile, die im Vergleich zu anderen Sprachen sehr ins Gewicht fallen können, vor allen Dingen bei kleineren Computeranlagen.
Neben diesen Vorteilen hat Assembler allerdings sehr schwerwiegende Nachteile aufzuweisen: Assembler-Programme sind nicht zu lesen, da sie nicht mit Befehlen geschrieben werden, welche sofort ihre Wirkung auf den Programmablauf erkennen lassen, sondern in Form eines Code.
An dieser Stelle sollte vielleicht einmal gesagt werden, was ein Befehl in einem Programm ist: Mit der Bezeichnung Befehl ist fast immer ein Makro-Befehl gemeint. Er enthält meist mehrere Schritte, die für den Pro-

grammierer allerdings nicht interessant sind.

Beispiel: Eine Addition in irgendeiner höheren Programmiersprache

A = B + C

In den meisten Programmiersprachen kann eine Addition in einer solchen Form angegeben werden.
Der Wert der Variablen B soll mit dem Wert der Variablen C addiert und das Ergebnis in der Variablen A gespeichert werden.
Für den Computer ist dies allerdings nicht ein Befehl, sondern gleich ein kleines Bündel von Befehlen:

1. Zunächst muß geprüft werden, wo sich im Speicher die Variablen B und C befinden. Die Stellen sind durch Adressen markiert.
2. Sind die Adressen gefunden, werden die Inhalte (Werte) von B und C gelesen.
3. Der Wert von B muß nun in das Addierwerk gebracht werden.
4. Der Wert von C muß ebenfalls in das Addierwerk übermittelt werden.
5. Dort wird der Wert von B und der Wert von C verrechnet und es entsteht ein neuer Zahlenwert, die Summe aus B und C.
6. Nun wird der Speicherplatz, der für die Variable A reserviert wurde, gesucht.
7. Der Inhalt des Addierwerkes muß nun in diesem Speicherplatz abgespeichert werden.

Will man das Ergebnis der Addition erfahren, so muß der Inhalt der Variablen A abgerufen werden.

In Assembler muß nun jeder Schritt für sich durchgeführt werden, wozu in anderen Sprachen eine kurze Zeile genügt. Außerdem sind andere Sprachen meist genormt, das bedeutet, die Anpassung einer bestimmten Sprache an einen bestimmten Computertyp geschieht mit Hilfe von Compilern oder Interpretern, die zur Ausstattung einer Anlage gehören. Da Assembler aber praktisch direkt mit den Maschinencodes arbeitet, gibt es für jeden Anlagentyp ein anderes Assembler als Sprache und ein anderes Assembler als Programm zur Umwandlung des Quelltextes in den Maschinencode.
Assembler gibt es seit 1953.

9. ATLAS
Anwenderspezifische Realzeitsprache.
ATLAS ist maschinenunabhängig und kann daher als „höhere" Programmiersprache bezeichnet werden, jedoch eignet sie sich vorwiegend zum Testen von Geräten, wozu sie auch eigentlich entwickelt wurde.

10. Autocode
Maschinenorientierte, sehr vereinfache Programmiersprache.
Autocode ist der Vorläufer der Assembler-Sprachen und wurde 1953 entwickelt.
Autocode war durch seine Einfachheit leicht erlernbar. Die Nachteile lagen in der sehr starken Orientierung an der entsprechenden Rechenanlage (im Gegensatz zur Problemorientierung), daher sind die Programme nicht von einer auf eine andere Rechenanlage gleichen Typs übertragbar (wie bei Assembler).
Aus Autocode entstand später Autocoder. Beide Sprachen sind heute kaum noch gebräuchlich. Assembler hat sich wegen seiner besseren Strukturen durchgesetzt.
Autocode war die Programmiersprache der zweiten Anlagengeneration (etwa 1955 bis 1965).

11. Autopit
Automatisches **P**rogrammieren **I**nclusive **T**echnologie.
Eigenentwicklung der Firma Pittler. Autopit hat deutsche Befehlsfolgen und einige Vorteile in der Materialbearbeitung zu bieten.
Das System von Autopit berücksichtigt Abmessungen und Materialbesonderheiten und trifft daraufhin automatisch Entscheidungen.
Autopit ist maschinenorientiert und kann auch bei weniger großen Anlagen sinnvoll eingesetzt werden, ist aber speziell für einen Maschinentypus der Firma Pittler entwickelt worden.

12. Autopol
Automated **P**rogramming **O**f **L**athes.
Für Drehbanksteuerung und andere Maschinen mit zweidimensionalen Bewegungsabläufen (siehe APT).
Trotz eines sehr kleinen Sprachumfangs können alle erforderlichen Vorgänge gesteuert werden, wodurch Autopol sehr benutzerfreundlich ist.
Das fertig erstellte Programm muß mit Hilfe eines Anpassungsprogrammes übersetzt werden (ähnlich dem Compiler- oder Assemblerprinzip).

13. AUTOPROG
Entwickelt in der CSSR für numerische Steuerung.
Anwendung nur regional.

14. Autopromt
Automated **Pro**gramming **O**f **M**achine **T**ools.
Für die Positionierung von Werkzeugen im Maschinenbau (NC-Steuerung).
Diese Sprache zeichnet sich aus durch ihren konkreten Bezug zu der Bezeichnungsweise in ihrem Anwendungsbereich, dem Maschinenbau. Es ist daher relativ einfach, anhand von technischen Zeichnungen das Steuerprogramm für die Werkzeugmaschinen zur Fertigung eines Werkstückes herzustellen (siehe APT).
Da die übliche Terminologie aus dem Maschinenbau in symbolischer Form in die Syntax von Autopromt übernommen wurde, ist diese Programmiersprache besonders benutzerfreundlich und hat auch selbstdokumentierenden Charakter. Autopromt ist daher für die Werkzeugmacher leicht erlernbar.

15. Autospot
Automated **S**ystem **F**or **P**osition **O**f **T**ools.
Für Bohr- und Drehmaschinensteuerung (siehe APT).
Auch Autospot-Programme müssen nach der Erstellung zunächst mittels eines Anpassungsprogramms übersetzt werden. Der vom Computer daraufhin erstellte Lochstreifen wird zur Maschinensteuerung benutzt (NC-Steuerung).

16. BASIC
Beginners **A**ll purpose **S**ymbolic **I**nformation **C**ode.
BASIC wurde ursprünglich entwickelt, um Anfängern den Einstieg in die Programmiertechnik zu vereinfachen, hat sich aber durch die vielseitigen Möglichkeiten der Ein- und Ausgabe von Daten zur Dialogsprache schlechthin entwickelt. BASIC zeichnet sich aus durch gute String-Verarbeitung (Zeichenketten, Texte) und einfach aufgebaute Befehle.

BASIC – PROGRAMMIERSPRACHE FÜR JEDERMANN

BASIC entstand etwa 1964 an der Dartmouth University in den USA unter John Kemeny und Thomas Kurtz.
Es dürfte die wohl bekannteste Programmiersprache sein, nicht zuletzt durch die Technologie der Mikrocomputersysteme, die heute etwa zu 90 Prozent mit BASIC und mit Dialekten von BASIC arbeiten können.
Die ersten Microcomputer (heute würde man sie als Homecomputer klassi-

fizieren) konnten zum Teil sogar nur in BASIC programmiert werden, weil der BASIC-Interpreter fest „verdrahtet" war (ROM).
BASIC ist eine dialogorientierte Sprache, was nicht heißt, daß mit BASIC keine mathematischen Probleme gelöst werden können, da große Ähnlichkeiten zu Fortran und Algol bestehen.
Da BASIC im Gegensatz zu den genannten anderen Programmiersprachen mit einem Interpreter arbeitet, ist das Programmieren, Fehlersuchen und Austesten von Programmen sehr komfortabel und schnell. Ein Editor wird beim Programmieren nicht benötigt, da die Programmzeilen direkt vom Bildschirm in den Interpreter übernommen werden. Noch vor der Ausführung des Programms (oder dessen Interpretation) wird jede Zeile bei der Eingabe auf Syntaxfehler untersucht.

Beispiel:
Eingabe der Zeile
15 PINT A
Meldung des Interpreters
? SYNTAX ERROR
Korrektur der Zeile
15 PRINT A

Der Anwendungsbereich von BASIC übergreift den vieler anderer Programmiersprachen.

```
Tabelle 6:

------------------------------------------------------------
Programmbeispiel zu Cobol
------------------------------------------------------------
PROCEDURE DIVISION
   OPEN INPUT FILES
   OUTPUT PRINTER
   READ FILE8
FOLLOWING
   IF NUMBER IN B IS EQUAL LAST
   GO TO END ELSE NEXT SENTENCE.
   MOVE NUMBER IN B TO NUMBER IN
   ACAT, INTER
CONTINUE
   ADD TOTAL TO S IN B
   GIVING TO TOTAL
   READ FILE8
   IF NUMMBER IN P IS EQUAL INTER
   GO TO CONTINUE, ELSE NEXT SENTENCE
   MOVE TOTAL TO GRAM IN ACAT
   WRITE ACAT
   GO TO FOLLOWING
END
   CLOSE FILE8,PRINTER
   STOP RUN
------------------------------------------------------------
```

SCHWIERIGE STRUKTURIERUNG

Der größte Nachteil von BASIC für den Programmierer ist die schlechte Lesbarkeit von umfangreicheren fertigen Programmen, da es keine strengen Strukturierungsmodi gibt, an die sich der Programmierer halten muß, wie in anderen Programmiersprachen. So ist die Gefahr der Verschachtelung von Programmen weitaus höher als etwa bei Pascal, Fortran oder Cobol.
Daher wird BASIC gerne nachgesagt, es forme beim Anwender einen schlechten Programmierstil und sei daher für Anfänger nur bedingt geeignet.
In BASIC müssen keine Variablennamen deklariert werden, jedoch sind die Namen nur zwei Zeichen lang, was die Lesbarkeit der Programme erschwert.
Die BASIC-Syntax selbst ist leicht verständlich und die meisten Befehle sind selbstdokumentierend.
Unterprogramme werden in BASIC nicht mit einem Label (Namen) versehen, sondern können direkt angesteuert werden. Nachteil: Fehlerquelle und Unübersichtlichkeit. Außerdem werden Parameter immer automatisch den Unterprogrammen zur Verfügung gestellt, was die Flexibilität eingeschränkt.
BASIC-Dialekte sind zum Beispiel: BASIC 80, Integer BASIC, Microsoft BASIC, BASIC 86, Advanced BASIC, XY-BASIC, EXTRABASIC, C-BASIC-2, Multiuser BASIC, Level III BASIC, Better BASIC, usw., die Reihe könnte beliebig fortgesetzt werden. Nur wenige BASIC-Versionen bringen wirklich interessante Neuerungen, wie GfA-BASIC: Hierbei handelt es sich um einen BASIC-Dialekt mit Pascal- und C-Elementen. GfA-BASIC arbeitet ohne Zeilennummern und kann compiliert und im Direktmodus angewendet werden, ähnlich dem True BASIC für den Amiga.

Die Standard-BASIC-Syntax:

Anweisungen
LET
GOTO
STOP
END
REM
IF..THEN
FOR..NEXT
GOSUB
RETURN
DIM
DEF
RESTORE

Standardfunktionen
SQR Wurzelfunktion
ABS Absolutfunktion
SGN Vorzeichenfunktion
LOG Logarithmusfunktion
EXP Exponentialfunktion
SIN Sinusfunktion
COS Cosinusfunktion
TAN Tangensfunktion
ATN Arcustangensfunktion
INT Ganzzahlfunktion
RND Zufallszahlenfunktion

Systembefehle
LIST
NEW
RUN
SAVE
LOAD

E/A Anweisungen
READ
OPEN
CLOSE
DATA
INPUT
GET
PRINT
PRINT USING

Operatoren
+ Addition
– Subtraktion
* Multiplikation
/ Division
↑ Potenzierung
AND Konjunktion
OR Disjunktion
NOT Negation

Obwohl BASIC starke Konkurrenz bekommen hat, ist es immer noch *die* Programmiersprache für den Anfänger. Wer autodidaktisch arbeitet, also sich selbst Programmierkenntnisse aneignen will, wird in der Regel auf BASIC zurückgreifen. Fast alle Homecomputer arbeiten standardmäßig mit BASIC-Dialekten.
BASIC-Interpreter benötigen wenig Speicherplatz und sind bei den meisten Home-Computern in ROMs (Read Only Memory) untergebracht, sonst auf Diskette verfügbar.
Zur Codeoptimierung gibt es auch BASIC-Compiler. Durch sie wird ein geringer Geschwindigkeitsgewinn erzielt. Im professionellen oder semi-professionellen Programmbereich ist BASIC selten zu finden.

Seit Mitte 1987 gibt es eine weitere Version mit dem Namen True BASIC (das wahre BASIC). Seiner Entwicklung geht eine Geschichte voraus: Viele BASIC-Dialekte, die aufgrund der eingeschränkten Möglichkeiten der verfügbaren Hardware entwickelt wurden, ließen eben auch nur eingeschränkte Möglichkeiten der Programmiersprache und der damit erstellten Software zu. Zu dieser engen Verknüpfung von BASIC mit der Hardware eines bestimmten Computers kam eine große Einschränkung der Portabilität, obwohl gerade diese ein Hauptkriterium für die Entwicklung von BASIC war.

TRUE BASIC: BESSER ALS DER STANDARD

Die hier genannten Nachteile, insbesondere die Einschränkungen aufgrund der Hardwarevoraussetzungen, wurden BASIC im allgemeinen nachgesagt, obwohl sie im Grunde nicht der Sprache an sich anzulasten waren. Die Entwickler hatten weder eine unübersichtliche, noch eine schwerfällige Sprache konzipiert. Daher wurde True BASIC geschaffen. Hiermit haben die ursprünglichen Entwickler von BASIC, Kurtz und Kemeny, versucht, die damaligen Vorstellungen von BASIC wieder zu realisieren, da heutige Microcomputer wesentlich bessere Voraussetzungen für die Realisation dieses Projektes besitzen. True BASIC entspricht vollkommen dem National Standard For BASIC und garantiert damit ein hohes Maß an Portabilität. Es ist ein Versuch der Rehabilitation dieser Sprache.

17. BCPL

BASIC Combined Programming Language. Assemblerähnliche Programmiersprache. Vorläufer von C (siehe dort).
1969 auf den Markt gebracht, verbindet BCPL die Struktur von Algol mit der Leistungsfähigkeit von Assembler.
BCPL wurde von Martin Richards am Massachusetts Institute of Technology entwickelt. Bis Ende der 60er Jahre wurde BCPL überwiegend und mit Erfolg bei der Systemprogrammierung eingesetzt.
Am Bell Laboratorium in New Jersey entwickelte Ken Thompson hieraus die typenlose Sprache B. BCPL war ebenfalls typenlos, wenn der Typ des Maschinenwortes unberücksichtigt bleibt.
Aus B, einer aus BCPL abgeleiteten problemorientierten Programmiersprache, wurde Mitte der 70er Jahre C entwickelt.

18. BLISS-32

Systemimplementierungssprache für VAX-Systeme. BLISS-32 unterstützt die Entwicklung modularer Software nach dem Konzept der strukturierten Programmierung.
Durch die Möglichkeit des direkten Zugriffes auf alle Hardware-Komponenten der VAX-Systeme wird damit die Programmierung von Echtzeit- und Hardware-abhängigen Anwendungen erleichtert.
BLISS-32 wird hauptsächlich zur Entwicklung von Betriebssystemen, Compilern, Laufzeitsystembausteinen, Datenbanksystemen, Kommunikationssoftware und Dienstprogrammen eingesetzt.
Die Version VAX-BLISS-32 erlaubt insbesondere die Entwicklung von systemnaher Software für die VAX. Die Basisfunktion der VAX-Architektur können direkt angesprochen werden.

19. C

Weiterentwicklung von BCPL.
C ist eine Systemimplementierungssprache. Neben der Programmierung von Betriebssystemen wird C aber auch in technisch-wissenschaftlichen Bereichen eingesetzt, da ihr Niveau dem von Algol, PL/1 oder Pascal angeglichen ist. C ist stark blockstrukturiert.
C gibt es seit 1972 und wurde von Dennis Ritchie aus der Systemimplementierungssprache B durch Erweiterungen entwickelt, zum Beispiel durch Einfügen von Datentypen.
Die Sprache C zeichnet sich vor allem durch ihre hohe Verarbeitungsgeschwindigkeit und ihre Maschinennähe trotz hohem Programmniveau aus. Der Grad der Komplexibilität des Programmcodes ist frei wählbar, wobei sich Lesbarkeit und Schreibaufwand natürlich konträr verhalten.
Weitere Vorteile liegen in der weniger strengen Abfassung als etwa PL/1, was aber zugleich einen großen Nachteil mit sich bringt: Die Fehlersuche wird erheblich erschwert und kompliziert hierdurch und durch einige andere Hindernisse das Erlernen der Sprache und deren Anwendung beträchtlich.

C: DIE NÄCHSTE GENERATION

C ist sicherlich eine wichtige Entwicklung auch für die zukünftigen Rechnergenerationen, mit Sicherheit aber keine Sprache für Anfänger.
C ist Hardware-unabhängig und daher universell einsetzbar.
Die Leistungsfähigkeit von C konnte zum ersten Mal 1972 unter Beweis gestellt werden. In den Bell Laboratories wurde zu dieser Zeit das Betriebssystem Unix vorgestellt. Dieses mittlerweile weit verbreitete Betriebssystem ist vollständig in C programmiert.
C ist heute die Standardsprache für professionelle Software-Entwicklungen in allen Bereichen. Kompaktheit, Effizienz und Schnelligkeit, sowie ein hohes Maß an Portabilität zwischen verschiedenen Hardware-Konfigurationen und Betriebssystemen zeichnen C aus.
Die Maschinennähe von C-Programmen – bei gleichzeitiger Hardware-Unabhängigkeit – läßt die Erstellung universell lauffähiger Programme zu. Zeitaufwendige Implementierungen für andere Rechnertypen können praktisch entfallen.
Die Hochsprache C ist sicherlich nichts für Anfänger. Um hier sinnvoll und routiniert mit den Sprachmitteln umgehen zu können, bedarf es einigen Aufwandes. Über die Grundsätze der Software-Entwicklung sollte man Bescheid wissen, bevor man sich mit C befaßt. Von Vorteil ist sicherlich die Kenntnis einer anderen höheren Programmiersprache.

SCHWIERIGE SYNTAX

Die Schwierigkeiten der Anwender resultieren unter anderem aus der schlechten Lesbarkeit von C-Programmen und der unter Umständen schwierigen Fehlersuche.
Die C-Syntax ist für Fehler geradezu prädestiniert, da manche Symbole je nach Zugehörigkeit zu anderen Operatoren auch andere Bedeutungen erhalten. Das Minuszeichen etwa bedeutet sowohl den zweistelligen Subtraktions-Operator, als auch das einstellige Minus. Viele Operationen bestehen zudem aus mehr als einem Zeichen. Eine Zeichenfolge kann, je nach Kontext, also auf die eine oder andere Art interpretiert werden. Die Zeile: A + + + B hat in C die Bedeutung (A + +) + B, könnte aber im Programm auch ganz anders gelesen werden, etwa als A + (+ + B), was in C durchaus zugelassen ist; jedoch liefern beide Zeilen verschiedene Ergebnisse, was vom Programmierer nicht gewollt sein kann.
Für Microcomputer gibt es bereits eine Reihe von

Implementierungen für C, so zum Beispiel:

Turbo-C von Borland und Heimsoeth,
Quick-C von Microsoft,
Zorland-C von Zorland,
Aztec-C von Manx.

Kaum hat sich C auf dem Markt etabliert, gibt es schon einen Nachfolger: C plus plus oder, um bei der Syntax von C zu bleiben C++.
Wie das alte C wurde auch C++ in den Bell Laboratories entwickelt, allerdings von Bjarne Stroustrup. C++ erschien zum ersten Mal 1983 unter dem Namen: C with Classes. C++ ist aufwärtskompatibel, das heißt, „alte" C-Programme laufen auch unter C++.
Dieses Konzept wurde zum ersten Mal konsequent bei Betriebssystemen durchgeführt. Bekanntestes Beispiel ist hier MS-DOS.

20. CDL/2
Systemimplementierungs-Sprache SIL (siehe SL/3). SIL-Sprachen sind zwar Code-effizient, aber recht unbequem in der Programmierung.

21. Cobol
Common **B**usiness **O**riented **L**anguage.
Sprache für den kaufmännischen Bereich.
Die Sprache Cobol hat ihre Stärken dort, wo Fortran Schwächen aufweist: in der Handhabung großer Datenmengen und dem Aufbau von Dateien.
Cobol gibt es seit Ende der 50er Jahre und wird ständig überarbeitet: vom **A**merican **N**ational **S**tandards **I**nstitute zu ANSI-Cobol oder von Codasyl, der **Co**nference of **Da**ta **Sy**stems **L**anguage.
Cobol wurde wie Ada im Auftrag des US-Verteidigungsministeriums entwickelt.
Codasyl ist ein Gremium zur Weiterentwicklung von Programmiersprachen, im besonderen von Cobol.
In Deutschland ist Cobol in der DIN 66028 festgelegt: „Informationsverarbeitung; Programmiersprache Cobol".
Cobol kommt besonders dort zum Einsatz, wo große Mengen von kommerziellen Daten verarbeitet werden sollen. Es gehört zu den sogenannten selbstdokumentierenden Programmiersprachen. Das bedeutet, die Programme sind leicht lesbar, was die teilweise umständliche Programmierung ausgleicht. Durch die guten selbstdokumentierenden Eigenschaften ist Cobol leicht erlernbar und auch von Nichtprogrammierern gut anzuwenden.
Cobol ist in starkem Maße rechnerunabhängig, also eine echte problemorientierte Sprache. Der Einsatz fand bisher meist auf größeren Anlagen statt. Heute gibt es jedoch leistungsfähige Compiler für viele Kleinrechner und Microcomputer. Ab 1985 beginnt Cobol, auch die MS-DOS-Welt auf dem IBM-PC zu erobern. Es gibt bereits sehr gute Compiler für IBM-kompatible Personal Computer. Diese Programme sind sogar portabel auf Großrechneranlagen.

Cobol-Programme bestehen immer aus vier *Programmteilen* (Divisions). Dadurch wird eine starke Strukturierung erreicht:

1. Erkennungsteil – Identification Division. Hier stehen Angabe zur Kennzeichnung des übersetzten Programmes in Form einer Liste.
2. Maschinenteil – Environment Division. Es muß angegeben werden, auf welcher Anlage dieses Programm laufen soll und welche peripheren Geräte vom Programm angesprochen werden. Hierunter fallen auch Dateien.
3. Datenteil – Data Division. Die Form der Daten, die vom Programm verarbeitet werden, muß eindeutig festgelegt werden, außerdem die Speicherbereiche und ob es sich um Variablen oder Konstanten handelt.
4. Prozedurteil – Procedure Division. Hier werden alle Abläufe und Operationen beschrieben, die ausgeführt werden sollen.

Im letzten Teil, dem Prozedurteil, steht das eigentliche Cobol-Programm. Die übrigen Divisions definieren lediglich den Programmnamen, die Peripherie (im Programm angesprochene Geräte wie Drucker, Floppy) und die Dateien und Speicherbereiche der Daten.
Das Standard-Cobol enthält folgende Befehle:

Operatoren:
ADD
SUBTRACT
MULTIPLY
DIVIDE
COMPUTE

Steueranweisungen:
IF
ALTER
PERFORM
GO
STOP

E/A-Anweisungen:
READ
WRITE
ACCEPT
DISPLAY
OPEN
CLOSE

Übersetzer-Anweisungen:
USE
ENTER
EXIT
NOTE

Daten-Übertragung:
MOVE
TRANSFORM
EXAMINE

In Cobol können sogenannte Wörter definiert werden. Ein Wort besteht aus maximal 30 Zeichen:
Zahlen 0 bis 9,
Zeichen A bis Z.

Beispiele:
Brutto-Lohn
Netto-Lohn
Personenkennziffer.

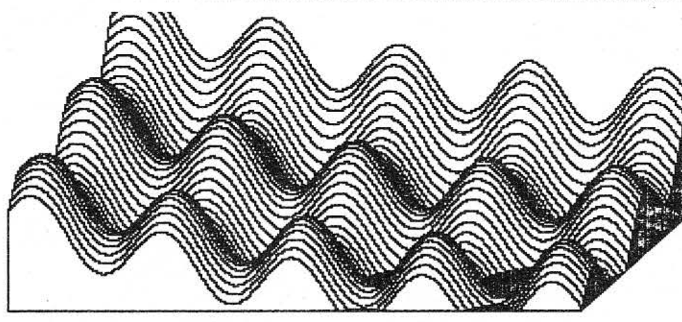
Leerzeichen dürfen nicht in Wörter geschrieben werden, denn sie dienen der Trennung.
Außer den frei wählbaren Wörtern gibt es die Cobol-Wörter. Sie sind bereits definiert und haben feste Bedeutungen. Es gibt drei Arten von Cobol-Wörtern:

1. Bindewörter
   Sie werden dazu verwendet, die Anwesenheit eines Kennzeichens anzuzeigen (In und Off), bzw. Boolsche Operatoren zu bilden (And, Or, Not . . .)
2. Wahlwörter
   Sie dienen der besseren Dokumentation der Programme und steigern so die Lesbarkeit. Das Wahlwort ändert nichts an der Interpretation einer Anweisung durch den Übersetzer, muß aber der vorgegebenen Syntax entsprechen.
3. Schlüsselwörter
   Sie sind notwendig zur Vollständigkeit einer Erklärung und müssen so verwendet werden, wie in den Formaten angegeben.

Wie in anderen Programmiersprachen gibt es zwischenzeitlich auch in Cobol einige Versionen und Dialekte.
Wie kaum eine andere Programmiersprache wurde Cobol erweitert und verbessert. Nur so ist zu erklären, daß sich Cobol über Jahrzehnte auf dem Markt behaupten konnte.
Hier die chronologische Entwicklung von Cobol:
Cobol 59
Cobol 60
Cobol 61
Cobol 63 (extended)
Cobol Edition 65
Cobol 68
Cobol 69
Cobol 70
Cobol 73
ANSI 74 Cobol
Cobol 80
ANSI 85 Cobol.
Die vorläufig letzte Version von Cobol, der ANSI-85-Standard, hat der betagten Programmiersprache einen neuen Charakter gegeben.
Die Programme sind übersichtlicher und die Cobol-Syntax wurde entrümpelt. Dennoch finden

Tabelle 7:

```
Programmbeispiel zu Comal 80
Listing
-------------------------------------------------------------
 10 DIM  ANTWORT$ OF 10
 20 PRINT "WEITER MIT START, CONT ODER STOP"
 30 INPUT ANTWORT$
 40 CASE ANTWORT$ OF
 50 // ES WURDE 'START' EINGEGEBEN
 60 WHEN "START"
       GOSUB PROC STARTEN
 70 // ES WURDE 'STOP' EINGEGEBEN
 80 WHEN "STOP"
       PRINT "DAS PROGRAMM WURDE BEENDET"; STOP
 90 // ES WURDE 'CONT' EINGEGEBEN
100 WHEN "CONT"
       PRINT "ES WIRD FORTGEFAHREN"
110 GO TO 200
120 // ES WURDE EINE FALSCHE EINGABE GEMACHT
130 OTHERWISE
140 PRINT "SIE HABEN EINE FALSCHE ANWEISUNG GEGEBEN"
150 GO TO 20
160 ENDCASE
200 // HIER WIRD FORTGEFAHREN
```



sich im ANSI 85 Cobol mehr als 40 neue Funktionen.
Es ist auch wichtig, daß sich eine Sprache der technischen Entwicklung anpaßt, denn vor 20 Jahren gab es noch keine Bildschirmterminals, so Bildschirmterminals. So fehlen in alten Programbebefehle für den Bildschirm. Überflüssig auf der anderen Seite sind Lochkartenbefehle – sie können heute aus einer Programmiersprache eliminiert werden. Diese Reihe könnte beliebig fortgesetzt werden und gilt natürlich nicht nur für Cobol.

22. Comal 80

Common **Al**gorithmic Language.
Comal wurde 1980 in Dänemark von Borge Christensen (State Teachers College) und Benedict Loftsted (Department für Computerwissenschaft der Universität Aarhus) entwickelt. Die Grundstruktur von Comal 80 ist der des BASIC gleich, jedoch um einige Elemente aus anderen Programmiersprachen, zum Beispiel Pascal, erweitert worden. Die Ähnlichkeiten mit BASIC sind allerdings wesentlich. Comal 80 ist somit als eine Erweiterung von BASIC anzusehen, wird dieses aber in Zukunft nicht verdrängen können.
Im Gegensatz zu BASIC und vielen anderen Programmiersprachen sind in Comal längere Variablennamen zulässig (maximal 16 Zeichen).
In Comal 80 ist es möglich, Unterprogramme zu benennen und Parameter zu übergeben, was in BASIC nicht machbar ist. Comal 80 ist mehr als BASIC selbst dokumentierend, behält aber alle Vorteile der Dialogsprachen.
Der Befehlsvorrat entspricht dem von BASIC in leicht modifizierter Form, zuzüglich der folgenden Anweisungen:

CASE .. ENDCASE
WHILE .. ENDWHILE
REPEAT .. UNTIL
IF .. THEN ..ELSE

23. Coursewriter

Spezielles System für computerunterstütztes Lernen (CAI=Computer-Aided Instruction).
Im Bereich der Ausbildung tritt der Computer mit Coursewriter-Programmen an die Stelle der Lehrperson.
CAI ist maschinenaufwendig, ebenso die Coursewriter-Programme. Vorteil der Methode ist, daß der Schüler das Lerntempo individuell bestimmen kann, und auch die Erfolge werden individuell erlebt.

DIE UNBEKANNTEN

Das Erlernen des Systems ist auf minimale Notwendigkeiten beschränkt, so bleibt dem Schüler mehr Raum für den eigentlichen Lerninhalt.
Der Nachteil hieraus wird durch den hohen Aufwand verursacht, die hohen Kosten, wodurch sich Coursewriter nicht durchsetzen konnte.

24. CSL

Control and Simulation Language.
Spezielle Programmiersprache zu Simulationszwecken und Modelluntersuchungen (siehe auch SIMULA).

25. DIPOL

Anwenderspezifische, problemorientierte Echtzeitsprache für die Automatisierung von Stückprozessen.

26. DL

Data **L**anguage.
Abkürzung für verschiedene Datenbanksprachen (zum Beispiel Query Language).

27. Elan

Extentab**l**e **La**ngua**g**e.
Elan ist wie C (siehe dort) eine Systemimplementierungssprache. Das Betriebssystem Eumel ist zum Beispiel in Elan geschrieben. Elan ist kaum verbreitet.

28. EOL

Expression **O**riented **L**anguage.
EOL wurde um 1960 von dem polnischen Mathematiker Jan Lukasiewicz entwickelt.
Aufsehen erregte sie durch die ungewöhnliche Notation zur Formulierung arithmetischer Ausdrücke. Diese Notation wurde zunächst nach ihrem Erfinder Lukasiewicz benannt. Aufgrund der schwierigen und ungewöhnlichen Schreibweise dieses Namens wurde diese Notation in den USA umbenannt in PN (Polish Notation).
Bei der PN werden immer zuerst die Operatoren, dann die zugehörigen Operanden geschrieben. Daher wird PN auch als Präfix-Schreibweise bezeichnet.

Beispiel:
5+4, – in PN: + 5 4

PN wird bei einigen Programmiersprachen auch umgedreht angewendet, also zuerst die Operanden und dann der oder die Operatoren. Diese Schreibweise bezeichnet man als RPN, also Reverse Polish Notation, bzw. Postfix-Schreibweise.

Beispiel:
5+4,– in RPN: 5 4 +

Bekannt wurde RPN durch die Taschenrechner von Hewlett Packard.

Dipl.-Ing. FH
Oliver Rosenbaum☐

Fortsetzung im nächsten Heft

Geschwindigkeit ist keine Hexerei

Wenn schon der Computer nicht schneller gemacht werden kann und auch Maschinensprache-Befehle böhmische Dörfer sind, muß eben das Programm beschleunigt werden. Mit dem Austrospeed+4 ist dies kein Problem.

Die bescheidene Geschwindigkeit von BASIC-Programmen ist manchmal ärgerlich, besonders bei zeitkritischen Befehlsfolgen wie Berechnungen oder Sortierungen. Das liegt einzig und allein daran, daß BASIC eine Interpretersprache ist. Jede BASIC-Zeile wird von einem Interpreter auf Syntaxfehler und dergleichen überprüft, umgesetzt und abgearbeitet.
Dabei geht gegenüber der Maschinensprache eine Menge Zeit verloren. Bei Variablen muß der Interpreter im Variablenspeicher nach dem Namen suchen, bevor der den Wert verarbeiten kann. Auch die Suche nach den entsprechenden Zeilennummern bei GOTO- oder GOSUB-Sprüngen, wobei er immer wieder bei Null anfängt, benötigt einige Zeit. Da selbst Profi-Programme compiliert werden, sollte auch der Hobby-Programmierer zu diesem Hilfsmittel greifen.
Ein Compiler beschreitet einen anderen Weg. Er compiliert das BASIC-Programm nach einem besonderen Code, der sehr maschinennah und dadurch um einiges schneller als das ursprüngliche BASIC-Programm ist. Die Nachteile des Interpreters fallen dadurch weg. Eine Überprüfung der Syntax findet nur noch einmal, beim Compilieren statt, wodurch die Ablaufgeschwindigkeit enorm gesteigert wird.

TUNING FÜR DAS 3.5-BASIC

Mancher Benutzer der 3.5-BASIC-Version (Plus4) wird schon neidisch auf andere Commodore-Besitzer geblickt haben, für deren Geräte es längst Compiler gibt.
Bereits seit einiger Zeit auf dem Markt, allerdings nur in Österreich, ist die Version des bekannten Austrocomp. Sie wird in der BRD von keinem uns bekannten Händler angeboten.
Man darf beim Kauf von Austrospeed+4 nicht verwundert sein, wenn er in der Verpackung des Austrocomp 64 versteckt ist. Der grundlegende Unterschied zwischen diesen beiden Compilern besteht nur in dem erweitertem Befehlsumfang der Plus4-Version. Darauf wird in einem Beiblatt hingewiesen, da auch die Gebrauchsanweisung auf den 64er-Compier zugeschnitten ist. Neben Handbuch und Programmdiskette wird ein sogenannter Dongle mitgeliefert. Das ist ein Stecker, der als Kopierschutz dient und auf den Userport gesteckt werden muß. Damit ist der Compiler nur auf dem Plus4 zu betreiben, da der C16/116 keinen Uersport besitzt.
Die Bedienung von Austrospeed+4 erfolgt über ein übersichtliches Menü, mit dem die entsprechenden Floppy-Konfigurationen eingegeben werden können. Zur Auswahl stehen die Möglichkeiten Singlefloppy, Doppellaufwerk oder zwei Einzellaufwerke. Bei einer Singlefloppy legt Austrocomp+4 das compilierte Programm wieder auf derselben Diskette ab, auf der sich das Ursprungsprogramm befindet. Man muß daher genügend Platz freihalten, da außerdem ein Zeilennummer-File abgelegt wird, das die Fehlersuche vereinfacht.
Austrospeed+4 kann auch ganze Programmpakete übersetzen. Ein Paket besteht aus einem Menüprogramm und mehreren unabhängigen Unterprogrammen, die von einem Menü aufgerufen werden. Austrospeed+4 compiliert die einzelnen Programme, versieht aber nur das Menüprogramm mit den zum ordnungsgemäßen Betrieb notwendigen RUN-Time-Modul. Von den Teilprogrammen erzeugt Austrospeed+4 lediglich den P-Code, der wesentlich kürzer ist und somit schneller nachgeladen wird.
Außerdem ermöglicht Austrospeed +4 die sogenannte Overlay-Technik, bei der nachgeladene Programme die Variablen aus vorhergehenden Programmen übernehmen.
Austrospeed+4 ist ein sogenannter 2-Pass-Compiler, er bearbeitet jedes BASIC-Programm in zwei Durchläufen: Im ersten werden alle Zeilennummern, auf die Sprungbefehle zeigen, markiert, übersetzt und mit den Sprungbefehlen gekoppelt. Dasselbe geschieht mit den Variablen. Im zweiten Durchlauf werden das RUN-Time-Modul, der P-Code und die Variablenlisten zu einem lauffähigen Programm zusammengefügt. Dadurch kann das erzeugte Programm größer werden, falls es vorher weniger als etwa 56 Blocks lang war. Alle Programme, die länger waren, werden nun etwas kürzer.

DER COMPILER ALS DEBUGGER

Da es recht mühsam ist, beim Übersetzen auftretende Fehler vom Bildschirm abzuschreiben, verlangt Austrospeed zwingend nach einem Drucker. Im anderen Fall wird beim Auftreten eines Fehlers die Fehlermeldung DEVISE NOT PRESENT ausgegeben. Damit kann Austrospeed+4 auch hervorragend als Debugger eingesetzt werden, denn die meisten Fehler werden beim Compilieren festgestellt und angezeigt.

DER DRUCKER IST EIN MUSS

Ist die Übersetzung erfolgreich verlaufen, zeigt Austrospeed+4 die Compilationsdauer und die Anzahl der Fehler an. Alle mit Austrospeed +4 übersetzten Programme sind auch ohne Compiler lauffähig und können, da sie eine BASIC-Zeile am Anfang haben, ohne Schwierigkeiten geladen und gesaved werden. Allerdings darf die BASIC-Zeile nicht nach eigenen Bedürfnissen abgeändert werden, da sonst das Programm nicht mehr läuft.

MEHR PLATZ AUF DEM STACK

Eine Besonderheit von Austrospeed +4 ist die erweiterte Verschachtelungstiefe bei GOSUB- und FOR-NEXT-Schleifen, wobei der gefürchtete STACK-Überlauf fast verhindert wird.
Alles in allem ist Austrospeed+4 ein sehr empfehlenswerter Compiler für den Plus4-Anwender, der nicht auf andere Produkte ausweichen kann. Eine Ausnahme ist zwar der Compiler von Markt & Technik. Er konnte aber nicht überzeugen. Bei der Überprüfung der Ausführungsgeschwindigkeit war Austrospeed+4 in fast allen Punkten zufriedenstellend. Bei reinen Grafikprogrammen müssen jedoch einige Abstriche gemacht werden.
Austrospeed+4 wurde ursprünglich in BASIC geschrieben und hat sich anschließend, wie viele Compiler, selbst übersetzt. *bw* □

Megatool

31 neue Befehle

Mit einer richtigen Syntax ausgestattet, bietet Mega-Tool eine Reihe nützlicher Funktionen an. So gibt es einen OLD-Befehl, einen Plot-Befehl und eine Merge-Routine, sogar Windows und Sprites sind spielend zu handhaben. Mega-Tool ist für jeden, dessen Rechner 64 KByte aufweist, eine wahre Bereicherung.

MEGA-TOOL ist eine BASIC-Erweiterung für den Plus4 und den C16 mit 64 KByte. Es erweitert das ohnehin umfangreiche BASIC V3.5 um weitere 31 Befehle, die die Programmierung dieses Computers sehr erleichtern.

OLD
Holt ein versehentlich gelöschtes Programm wieder zurück. OLD hilft manchmal auch, ein verstümmeltes Listing zurückzuholen.

LLIST g,s [,zeilen]
g. . . . . . . Geräteadr.
s. . . . . . . Sekundäradr.
zeilen . . wie bei LIST
Gibt Listing am Drucker aus.

MERGE "name",g
Verkettet zwei BASIC-Programme, wobei die Zeilennummern des nachgeladenen Programms höher sein müssen als die des ersten.

DMERGE "name"
Wie MERGE, nur auf Diskette.

BLOCK "name",g
Lädt einen Speicherbereich, ähnlich wie LOAD"name",g,1, allerdings mit dem Vorteil, daß der Programmzähler nicht zurückgesetzt wird, das Programm also nicht wieder von vorne beginnt. BLOCK kann zum Beispiel zum Nachladen eines Zeichensatzes verwendet werden.

DBLOCK "name"
Wie BLOCK, nur auf Diskette.

SBLOCK "name",g,1, anfang,ende+1
anfang. . . Anfangsadr.
ende. . . . Endadr.
Speichert einen bestimmten Speicherbereich auf Kassette oder Diskette ab. Kann zum Beispiel zum Abspeichern von Grafikseiten verwendet werden: SBLOCK "graphik",8,1,3072, 16384

DVERIFY "name"
Verifiziert auf Diskette.

ID
Initialisiert eine Diskette.

FAST
Schaltet den Bildschirm aus, wodurch der Computer um zirka 30 Prozent schneller arbeitet, da der TED-Chip nicht mehr auf das RAM zugreifen muß.

SLOW
Schaltet den Bildschirm wieder ein.

MULTI 0/1
Steuert das Multicolor-Bit des TED-Chips: MULTI 1 schaltet den Multicolor-Modus ein, MULTI 0 schaltet ihn wieder aus.

REVERS 0/1
Steuert das Revers-Flag des TED-Chips. Ist dieses Bit gesetzt, so erzeugt der TED keine reversen Zeichen und man kann auf den gesamten Zeichenvorrat des Computers zugreifen (statt revers Kleinschrift).

BIT adr,bit-#,0/1
adr. . . . . Speicheradr.
bit-# . . . Stellung des Bit (0–7)
Ermöglicht ein bitweises Poken.
Zum Beispiel entspricht BIT65298,2,0 dem Befehl POKE65298,PEEK (65298)AND255–4

TRANS 0/1
TRANS 1 kopiert den ROM-Zeichensatz ins RAM und zwar ab der Speicheradresse $E800, wo er verändert werden kann. Alle TED-Register werden auf diesen Zeichensatz umgestellt. Um zu verhindern, daß er von BASIC überschrieben wird, wird die Speichergrenze um ein KByte nach unten gesetzt. Dadurch werden alle Variablen gelöscht.

SPRITE nummer,x,y,col
nummer . . Spritenummer (0–15)
x,y. . . . . . Koordinaten des Sprites
col. . . . . . Farbe des Sprites (Bildschirmcode)
Der SPRITE-Befehl ist einer der mächtigsten Befehle dieser Erweiterung. Er dient zur Verwaltung

von 16 verschiedenen Pseudosprites, allerdings nur in Koppelung mit einem veränderten Zeichensatz.
Die Struktur der Sprites ist folgendermaßen:
Jedes Sprite besteht aus 2 mal 2 Zeichen.
Sprite Nummer 0 besteht aus den Zeichen 64, 65, 66 und 67, Sprite Nummer 1 aus 68, 69, 70 und 71 und so weiter, bis zu Sprite Nummer 15 aus den Zeichen 124, 125, 126 und 127.
Mit dem Zeichen 64 ist nicht das ASCII-Zeichen Nummer 64 (Klammeraffe) gemeint, sondern der Bildschirmcode Nummer 64, also das Sternchen mit SHIFT. Die vollständige Tabelle steht auf den Seiten 213 und 214 im Handbuch. Da fast alle Zeichensatz-Programme diesen Bildschirmcode verwenden (etwa der Graphic-Designer von Kingsoft), dürften sich damit kaum Probleme ergeben.
Das erste Zeichen des Sprites befindet sich links oben, das zweite rechts oben, das dritte links unten und das vierte rechts unten.
Wem ein Sprite aus vier Zeichen zu klein ist, muß mehrere Sprites zu einem zusammenfassen.
Zur Anwendung des Befehls: X ist die Spalte, Y die Zeile, in welcher das Sprite erscheint.
COL ist die Farbe, wobei Werte über 127 ein Blinken hervorrufen.
COL kann man sich mit folgender Formel errechnen: Farbe + Luminanz * 16.
Mit dem SPRITE-Befehl können Sprites neu gesetzt und bewegt werden. Allerdings sollte man es vermeiden, Sprites übereinander zu legen, da dies bei falscher Anwendung (Bewegen des darunterliegenden Sprites) unerwünschte Folgen haben könnte.
Der Status eines Sprites wird mit der USR-Funktion abgefragt: Ist der Wert größer als 128, so ist das Sprite am Bildschirm. Ist das sechste Bit gesetzt, so kollidierte es mit dem Untergrund: Es traf auf irgend ein anderes Zeichen als das Leerzeichen; andere Sprites gelten nicht als Hindergrund. Sollte das Sprite mit einem anderen kollidiert sein, so liefert USR(Spritenummer) AND 15 dessen Nummer.

DEL nummer
Löscht ein Sprite vom Bildschirm.

PLOT x,y
Setzt den Cursor an die Koordinaten x,y.

WINDOW x1,y1,x2,y2
x1,y1 .. linke obere Ecke
x2,y2 .. rechte untere Ecke
Definiert Fenster.

SCREEN x TO y
x .. Quellbildschirm
y .. Zielbildschirm
Der SCREEN-Befehl dient zum Verwalten der beiden RAM-Bildschirme.
Nummern:
0 . . .sichtbarer Bildschirm
1 . . .RAM-Bildschirm ab $EC00
2 . . .RAM-Bildschirm ab $F400
Mit SCREEN kann man den Bildschirm x in den Bildschirm y kopieren.
Als Anwendung wäre das Verwalten von Fenstern, wie beim Amiga, denkbar: Sichtbaren Bildschirm ins RAM kopieren, mit CHAR das Fenster bilden, zum Schließen des Fensters den Bildschirm wieder zurückkopieren. Ebenfalls möglich ist das Einblenden von Informationen wie Anleitungen oder Hi-Score-Tabellen.
Beim SCREEN-Befehl werden alle Bildschirmparameter wie Cursorposition, Revers-Flag, Hintergrund- und Rahmenfarbe mitkopiert.

HARDCOPY g,s
g,s . . . wie bei LLIST
Druckt den momentanen Bildschirm mit allen Grafikzeichen, sofern sie am Drucker vorhanden sind, aus.

INIT
Initialisiert Bildschirm, TED und Sprungvektoren, ohne ein Programm zu unterbrechen.

SWAP a$,b$
a$,b$ . . . beliebige Stringvariablen
Vertauscht zwei Stringvariablen ohne die Erzeugung von „Stringmüll".

SLEEP dauer
dauer . . . Zeit in Hundertstel Sekunden
SLEEP ersetzt die FOR-NEXT-Warteschleifen.
Der Befehl wird durch den Raster-Interrupt gesteuert und kann durch die RUN/STOP-Taste unterbrochen werden.
Ein Tip: Mit SLEEP 1 vor dem SPRITE-Befehl wird, vor allem bei sehr schnellen Spielen, Bildschirmflimmern unmöglich gemacht, da nach einem SLEEP-Befehl immer ein neuer Bildschirmaufbau stattfindet.

Lesen Sie bitte weiter a.S. 49

```
MEGA-TOOL========================c16
(p) COMMODORE WELT TEAM
====================================
(c) by Bernhard Oemer

C16/116/Plus4
====================================
```

Das Programm ist mit dem Maschinensprachmonitor TEDMON unter Zuhilfenahme des Pruefsummenprogrammes CHECKMON einzugeben und auf Diskette mit s"mega-tool",8,1001,17e8 abzuspeichern. Auf Kassette bitte mit s"manager",1,1001,17e8 abspeichern. Geladen und gestartet wird wie bei einem normalen Basic-Progamm.

```
====================================
>1000 00 0e 10 c3 07 9e 20 28 :<da>
>1008 34 31 31 32 29 00 00 00 :<9f>
>1010 a9 18 85 2c a9 00 8d 00 :<3a>
>1018 18 a9 ec 85 38 a2 05 bd :<d4>
>1020 96 10 9d 0c 03 ca 10 f7 :<c2>
>1028 20 81 ff 20 8a ff a9 7b :<5e>
>1030 8d 2a 03 a9 16 8d 2b 03 :<e2>
>1038 20 7b 16 a9 f0 8d 15 ff :<a8>
>1040 8d 19 ff 20 4f ff 0d 05 :<e7>
>1048 93 20 43 4f 4d 4d 4f 44 :<4b>
>1050 4f 52 45 20 42 41 53 49 :<7c>
>1058 43 20 56 33 2e 35 20 2b :<35>
>1060 20 4d 45 47 41 2d 54 4f :<08>
>1068 4f 4c 20 56 31 38 2e 30 :<6f>
>1070 0d 0d 20 20 33 30 20 4e :<61>
>1078 45 57 20 43 4f 4d 4d 41 :<10>
>1080 4e 44 53 20 41 4e 44 20 :<24>
>1088 00 20 e5 80 20 4f ff 0d :<2b>
>1090 0d 9e 00 4c 0a 80 9c 10 :<7b>
>1098 b0 10 d2 10 85 da a0 e1 :<c5>
>10a0 a9 10 20 07 8a 90 04 48 :<82>
>10a8 4c d6 89 a5 da 4c 6c 89 :<ea>
>10b0 aa a0 e1 84 22 a0 10 84 :<d6>
>10b8 23 a0 00 ca 10 10 b1 22 :<94>
>10c0 85 03 e6 22 d0 02 e6 23 :<0d>
>10c8 a5 03 10 f2 30 ed 18 4c :<e9>
>10d0 88 8b 29 7f 0a a8 b9 87 :<97>
>10d8 11 48 b9 86 11 48 4c 73 :<23>
>10e0 04 4f 4c c4 42 4c 4f 43 :<a2>
>10e8 cb 44 42 4c 4f 43 cb 4d :<f5>
>10f0 45 52 47 c5 44 4d 45 52 :<06>
>10f8 47 c5 53 50 52 49 54 c5 :<bf>
>1100 44 45 cc 46 41 53 d4 53 :<58>
>1108 4c 4f d7 4d 55 4c 54 c9 :<09>
>1110 52 45 56 45 52 d3 54 52 :<46>
>1118 41 4e d3 53 4c 45 45 d0 :<81>
>1120 50 4c 4f d4 57 49 4e 44 :<15>
>1128 4f d7 53 57 41 d0 42 49 :<26>
>1130 d4 52 45 53 45 d4 48 41 :<aa>
>1138 52 44 43 4f 50 d9 53 42 :<f1>
>1140 4c 4f 43 cb 4c 4c 49 53 :<e9>
>1148 d4 44 56 45 52 49 46 d9 :<a9>
>1150 53 43 52 45 45 ce 49 4e :<82>
>1158 49 d4 49 c4 57 41 49 54 :<ef>
>1160 4b 45 d9 42 55 46 46 45 :<1e>
>1168 d2 52 45 50 45 41 d4 52 :<28>
>1170 45 47 49 53 54 45 d2 44 :<7f>
>1178 49 53 cb 55 53 45 d2 00 :<48>
>1180 00 00 00 00 00 00 c3 11 :<4b>
>1188 ce 11 d4 11 e4 11 ea 11 :<4d>
>1190 16 12 a9 12 f5 12 fe 12 :<bd>
>1198 15 13 27 13 32 13 87 13 :<9f>
>11a0 cf 13 e7 13 18 14 43 14 :<4a>
>11a8 77 14 80 14 f7 14 0c 15 :<1f>
>11b0 37 15 52 15 e1 15 0c 16 :<44>
>11b8 27 16 58 16 86 16 0d 17 :<e1>
>11c0 26 17 ba 17 a9 01 a8 91 :<ef>
>11c8 2b 20 18 88 4c 4b 88 20 :<c2>
>11d0 4f 17 4c dc 11 20 56 17 :<39>
>11d8 a9 01 85 ad a2 ff a0 ff :<8a>
>11e0 a9 00 4c d5 ff 20 4f 17 :<a6>
>11e8 4c ee 11 20 56 17 a5 2b :<57>
>11f0 85 da a5 2c 85 db a5 2d :<44>
>11f8 a4 2e 38 e9 02 b0 01 88 :<e5>
>1200 aa a9 00 20 d5 ff a5 da :<d2>
>1208 85 2d a5 db 85 2c 20 c4 :<36>
>1210 11 20 18 88 4c dc 8b a9 :<9f>
>1218 08 8d 01 05 a9 13 8d 02 :<e0>
>1220 05 a9 00 85 dc 85 dd 20 :<64>
>1228 aa 12 20 d8 9d e0 27 b0 :<b7>
>1230 4d 86 db 20 d8 9d e0 18 :<47>
>1238 b0 44 86 d5 20 6a 17 20 :<7c>
>1240 d8 9d 86 d4 a5 da 0a 0a :<77>
>1248 aa a9 00 85 dc 85 dd a4 :<c2>
>1250 db 20 81 12 c8 20 81 12 :<0e>
>1258 98 18 69 27 a8 20 81 12 :<5d>
>1260 c8 20 81 12 a5 dc 09 80 :<45>
>1268 a6 dd f0 02 09 40 a6 da :<21>
>1270 9d 8b 17 a5 db 9d 9b 17 :<b8>
>1278 a5 d5 9d ab 17 60 4c 1c :<af>
>1280 99 b1 d0 c9 40 90 0a 29 :<b8>
>1288 3c 4a 4a 85 dc b1 d0 d0 :<34>
>1290 06 c9 20 f0 02 e6 dd 9d :<6d>
>1298 80 fc b1 d2 9d c0 fc a5 :<23>
>12a0 d4 91 d2 8a 09 40 91 d0 :<64>
>12a8 e8 60 20 84 9d e0 10 b0 :<48>
>12b0 cd 86 da bd 8b 17 30 01 :<a2>
>12b8 60 a9 00 9d 8b 17 bc 9b :<72>
>12c0 17 bd ab 17 aa 20 6a 17 :<50>
>12c8 a5 da 0a 0a aa 20 e3 12 :<d5>
>12d0 e8 c8 20 e3 12 98 18 69 :<93>
>12d8 27 a8 e8 20 e3 12 e8 c8 :<29>
>12e0 4c e3 12 78 8d 3f ff bd :<c0>
>12e8 80 fc 91 d0 bd c0 fc 91 :<3b>
>12f0 d2 8d 3e ff 58 60 ad 06 :<7d>
>12f8 ff 29 ef 8d 06 ff 60 ad :<ab>
>1300 06 ff 09 10 8d 06 ff 60 :<45>
>1308 20 87 9d e0 10 b0 1e bd :<b6>
>1310 8b 17 a8 4c 81 9a a9 10 :<7a>
```

```
>1318 85 da a9 07 85 14 a9 ff :<07>
>1320 85 15 20 84 9d 4c 57 14 :<d2>
>1328 a9 80 4c 18 13 4c 1c 99 :<eb>
>1330 4c 81 86 20 84 9d e0 01 :<39>
>1338 f0 07 e0 00 f0 37 4c 2d :<90>
>1340 13 a9 e8 85 38 20 9d 8a :<af>
>1348 a0 00 b9 00 d0 99 00 e8 :<2a>
>1350 b9 00 d1 99 00 e9 b9 00 :<e0>
>1358 d2 99 00 ea b9 00 d3 99 :<aa>
>1360 00 eb c8 d0 e5 a9 fb 2d :<20>
>1368 12 ff 8d 12 ff a9 e8 8d :<1c>
>1370 13 ff 4c dc 8b a9 ec 85 :<20>
>1378 38 20 9d 8a a9 04 0d 12 :<99>
>1380 ff 8d 12 ff a9 d0 d0 e7 :<9a>
>1388 20 e1 9d a5 14 d0 05 a5 :<f0>
>1390 15 d0 01 60 a5 15 30 95 :<6b>
>1398 a9 00 85 da 78 a9 be 8d :<54>
>13a0 14 03 a9 13 8d 15 03 58 :<95>
>13a8 a5 da d0 05 20 e1 ff d0 :<ea>
>13b0 f7 78 a9 0e 8d 14 03 a9 :<d1>
>13b8 ce 8d 15 03 58 60 c6 14 :<d5>
>13c0 d0 08 c6 15 10 04 a9 ff :<9b>
>13c8 85 da 4c 0e ce 4c 1c 99 :<a5>
>13d0 20 70 de 20 84 9d e0 28 :<34>
>13d8 b0 f3 86 ca 20 d8 9d e0 :<db>
>13e0 19 b0 ea 86 cd 4c a8 d8 :<95>
>13e8 20 84 9d e0 27 b0 de 8e :<44>
>13f0 e7 07 20 d8 9d e0 18 b0 :<d4>
>13f8 d4 8e e6 07 20 d8 9d 8a :<6c>
>1400 cd e7 07 90 c8 8d e8 07 :<e3>
>1408 20 d8 9d 8a cd e6 07 90 :<4a>
>1410 bc 8d e5 07 a9 93 4c 49 :<8b>
>1418 dc 20 2c 93 20 1a 93 a5 :<18>
>1420 64 85 d0 a5 65 85 d1 20 :<54>
>1428 91 94 20 2c 93 20 1a 93 :<fb>
>1430 a0 00 b1 d0 85 da b1 64 :<bb>
>1438 91 d0 a5 da 91 64 c8 c0 :<ee>
>1440 03 d0 ef 60 20 d2 9d e0 :<89>
>1448 08 b0 82 8a 49 07 aa bd :<d4>
>1450 89 c2 85 da 20 d8 9d a0 :<de>
>1458 00 b1 14 e0 01 f0 07 e0 :<35>
>1460 00 f0 08 4c 7e 12 05 da :<8f>
>1468 91 14 60 48 a5 da 49 ff :<c5>
>1470 85 da 68 25 da 91 14 60 :<7a>
>1478 20 2b cd f0 01 60 4c f9 :<d8>
>1480 ff 08 a9 00 85 ad a9 04 :<ff>
>1488 85 ae 28 f0 16 20 84 9d :<3c>
>1490 86 ae 20 73 04 f0 0c c5 :<d6>
>1498 2c f0 03 4c a1 94 20 84 :<17>
>14a0 9d 86 ad a9 00 85 ac 85 :<d7>
>14a8 ab 20 85 a7 a2 00 20 97 :<09>
>14b0 a7 a9 0d 20 8b a7 a2 00 :<9a>
>14b8 20 6a 17 a0 00 b1 d0 29 :<70>
>14c0 7f c9 20 b0 05 69 40 4c :<1f>
>14c8 d9 14 c9 40 90 0b c9 60 :<8c>
>14d0 b0 04 69 20 d0 03 18 69 :<38>
>14d8 40 20 8b a7 c8 c0 28 90 :<ef>
>14e0 dc a9 0d 20 8b a7 e8 e0 :<f0>
>14e8 19 90 cd a9 0d 20 8b a7 :<59>
>14f0 20 cc ff a9 00 4c c3 ff :<a8>
>14f8 20 6b a8 20 de 9d 84 d0 :<60>
>1500 85 d1 20 de 9d a6 14 a8 :<8f>
>1508 a9 d0 4c d8 ff 20 84 9d :<05>
>1510 86 ae 20 d8 9d 86 ad a9 :<a0>
>1518 00 85 ac 85 ab 20 85 a7 :<90>
>1520 a2 00 20 97 a7 a9 0d 20 :<27>
>1528 8b a7 20 79 04 f0 03 20 :<91>
>1530 91 94 20 ff 8a 4c eb 14 :<7e>
>1538 a9 e6 20 21 cb 20 b5 cc :<c8>
>1540 a2 00 8e 78 02 e8 85 0a :<b9>
>1548 a0 05 20 3f ca 4c fa a7 :<2a>
>1550 4c 1c 99 20 84 9d e0 03 :<f6>
>1558 b0 f6 86 da a9 a4 20 93 :<9a>
>1560 94 20 84 9d 8a c9 03 b0 :<f3>
>1568 e7 c5 da f0 e3 85 db a2 :<a3>
>1570 0b b5 c2 9d e8 0f ca 10 :<39>
>1578 f8 a2 03 bd ee 07 9d f4 :<95>
>1580 0f ca 10 f7 ad 15 ff 8d :<a6>
>1588 f8 0f ad 19 ff 8d f9 0f :<1f>
>1590 a9 00 85 d0 85 d2 a6 da :<5c>
>1598 bd 88 17 85 d1 a6 db bd :<79>
>15a0 88 17 85 d3 78 8d 3f ff :<41>
>15a8 a2 07 a0 00 b1 d0 91 d2 :<eb>
>15b0 c8 d0 f9 e6 d1 e6 d3 ca :<54>
>15b8 10 f0 8d 3e ff 58 ad f9 :<7d>
>15c0 0f 8d 19 ff ad f8 0f 8d :<7f>
>15c8 15 ff a2 03 bd f4 0f 9d :<35>
>15d0 ee 07 ca 10 f7 a2 0b bd :<1f>
>15d8 e8 0f 95 c2 ca 10 f8 4c :<f6>
>15e0 a8 d8 20 81 ff 20 84 ff :<4f>
>15e8 20 8a ff 20 cc ff 20 4f :<e2>
>15f0 ff 1b 4c 9e 93 00 a2 00 :<e3>
>15f8 8e 15 ff 8e 19 ff 86 de :<70>
>1600 a9 7b 8d 2a 03 a9 16 8d :<e5>
>1608 2b 03 4c 38 c7 a9 00 a2 :<1c>
>1610 08 a0 0f 20 ba ff a9 01 :<f0>
>1618 a2 27 a0 16 20 bd ff 20 :<cb>
>1620 85 a7 a9 00 4c c3 ff 49 :<4e>
>1628 20 7b 17 a5 61 f0 49 a0 :<f2>
>1630 00 78 8d 3f ff b1 62 8d :<3a>
>1638 3e ff 58 85 da 84 ef a5 :<e8>
>1640 91 c9 7f d0 01 60 a5 ef :<12>
>1648 f0 f5 78 a9 00 85 ef ad :<ce>
>1650 27 05 58 c5 da d0 e8 58 :<2b>
>1658 60 20 7b 17 a4 61 c0 0b :<05>
>1660 b0 16 84 ef 88 30 10 78 :<7a>
>1668 8d 3f ff b1 62 99 27 05 :<f7>
>1670 88 10 f8 8d 3e ff 58 60 :<64>
>1678 4c 1c 99 a2 ff 86 d6 86 :<97>
>1680 d7 e8 86 de 4c 08 ef 08 :<48>
>1688 20 86 9a 28 f0 3a c9 fd :<f0>
>1690 f0 6b c9 fc f0 6d 24 de :<a5>
>1698 10 0b a4 df f0 18 88 f0 :<ce>
>16a0 1c 84 df d0 23 20 e1 9d :<10>
>16a8 a5 14 85 df a5 15 85 e0 :<7f>
>16b0 a9 80 85 de d0 12 c6 df :<42>
```

```
>16b8 c6 e0 4c c8 16 84 df a4 :<58>
>16c0 e0 d0 05 84 de 4c e1 9d :<97>
>16c8 a5 d6 c5 39 d0 0f a5 d7 :<8a>
>16d0 c5 3a d0 09 a5 d8 85 3b :<cf>
>16d8 a5 d9 85 3c 60 a5 39 85 :<31>
>16e0 d6 85 14 a5 3a 85 d7 85 :<58>
>16e8 15 20 3d 8a a5 5f e9 01 :<77>
>16f0 85 3b 85 d8 a5 60 e9 00 :<5c>
>16f8 85 3c 85 d9 60 20 4c b6 :<55>
>1700 d0 c6 60 20 4c b6 f0 c0 :<c8>
>1708 60 a2 17 4c 86 86 20 84 :<ec>
>1710 9d 8e f2 07 20 d8 9d 8e :<5c>
>1718 f3 07 20 d8 9d 8e f4 07 :<a3>
>1720 20 d8 9d 8e f5 07 60 20 :<52>
>1728 7b 17 a4 61 f0 1e c9 28 :<b3>
>1730 b0 d7 a9 00 a2 08 a0 0f :<02>
>1738 20 ba ff a5 61 a6 62 a4 :<f3>
>1740 63 20 bd ff 20 85 a7 a9 :<3a>
>1748 00 4c c3 ff 4c 1c 99 a9 :<84>
>1750 00 85 0a 4c 6b a8 a9 e6 :<04>
>1758 20 21 cb 20 b5 cc a9 00 :<85>
>1760 8d 78 02 85 0a a0 05 4c :<81>
>1768 3f ca bd 02 d8 85 d0 85 :<05>
>1770 d2 bd 1b d8 85 d1 29 0b :<70>
>1778 85 d3 60 a5 16 48 20 2c :<62>
>1780 93 20 1a 93 68 85 16 60 :<c3>
>1788 08 ec f4 00 00 00 00 00 :<db>
>1790 00 00 00 00 00 00 00 00 :<38>
>1798 00 00 00 00 00 00 00 00 :<48>
>17a0 00 00 00 00 00 00 00 00 :<58>
>17a8 00 00 00 00 00 00 00 00 :<68>
>17b0 00 00 00 00 00 00 00 00 :<78>
>17b8 00 00 00 20 4f ff 0d 28 :<a5>
>17c0 43 29 20 42 59 20 42 45 :<3c>
>17c8 52 4e 48 41 52 44 20 4f :<fb>
>17d0 45 4d 45 52 20 28 31 39 :<2b>
>17d8 38 37 20 41 55 53 54 52 :<af>
>17e0 49 41 29 0d 00 4c 03 80 :<99>
==================================
P R O G R A M M E N D E
==================================
```

Fortsetzung von Seite 46

WAITKEY a$
Wartet, bis die Taste a$ gedrückt wird und entleert vorher und nachher den Tastaturpuffer.

BUFFER a$
Belegt den Tastaturpuffer mit dem String a$, wobei a$ nicht länger als 10 Zeichen sein darf.
Der BUFFER-Befehl kann vor allem zur Selbstmodifikation von Programmen verwendet werden.

REGISTER a,x,y,s
Definiert die Prozessor-Register vor einem SYS-Aufruf
a = Akkumulator
x = X-Register
y = Y-Register
s = Statusregister.

DISK a$
Sendet einen Befehl an die Diskettenstation
a$ = Befehlsstring.

REPEAT x / UNTIL cond / WHILE cond
Dieser Befehl dient zur Programmierung von kurzen Schleifen. Er steht am Ende einer Zeile.
Bei REPEAT x wird diese Zeile x-mal wiederholt, bei REPEAT UNTIL cond wird die Zeile wiederholt, bis die Bedingung cond erfüllt ist. Bei REPEAT WHILE cond wird die Zeile wiederholt, solange die Bedingung erfüllt ist.

RESET
Löst ein RESET aus. Im Direktmodus muß man vorher die Sicherheitsabfrage ARE YOU SURE? mit Y beantworten.

USER
Ruft die USER-Routine ($17BB) auf. Dort kann zum Beispiel der Aufruf eines eigenen MS-Programmes stehen. Nach dem Start der Erweiterung befindet sich an dieser Stelle die Copyright-Meldung.

FUNKTIONSWEISE DES PROGRAMMS:

Die Erweiterung liegt ab $1000 im Speicher. Nach dem Starten mit RUN wird der BASIC-Anfang auf $1800, das BASIC-Ende auf $EC00 gesetzt, um ein Überschreiben der Erweiterung, des Sprite-Speichers und der RAM-Bildschirme zu verhindern.
Außerdem werden die Vektoren zur Handhabung des USER-Tokens ($030C bis $0311) auf eigene Routinen gelegt. Dann werden der Bildschirm und der Speicher gelöscht, und MEGA-TOOL meldet sich mit der Einschaltmeldung.
Durch die Verwendung des USER-Tokens werden auch die neuen Befehle in Tokens umgewandelt und sind daher nur zwei Byte lang.
Als Arbeitsspeicher verwendet MEGA-TOOL den Bereich $EC00 bis $FCFF sowie die Zeropage-Adressen $D0 bis $D5, allerdings nur während der Ausführung des Befehls. Der übrige Systemspeicher bleibt unberührt, kann also für weitere Maschinensprache-Programme verwendet werden. Die hochauflösende Grafik steht uneingeschränkt zur Verfügung. □

Speicherplan von MEGA-TOOL:

$1001 – $100F
BASIC-Kopf
$1010 – $1095
Initialisierung
$1096 – $10E0
Behandlung des Tokens
$10E1 – $1180
Befehlswörter (Tokens)
$1186 – $11C3
Befehlsadressen
$11C4 – $1782
Routinen der Befehle
$1783 – $17BA
Arbeitsspeicher
$17BB – $17E7
USER-Routine

Abkürzungen der Befehle:

old	oL
block	bL
dblock	dB
merge	mE
dmerge	dM
sprite	sP
del	dE
fast	fA
slow	sL
multi	mU
revers	rE
trans	tR
sleep	sL
plot	pL
window	wI
swap	sW
bit	bI
reset	reS
hardcopy	hA
sblock	sB
llist	lL
dverify	dV
screen	sC
init	iN
id	id
waitkey	wA
buffer	bU
repeat	reP
register	reG
disk	dI
user	usE

Die Abkürzungen der herkömmlichen BASIC-Befehle können sich ändern: Die Abkürzung „scR, für „scratch", bedeutet mit MEGA-TOOL „screen".

Grafix 3D in Bewegung

Mit Grafix können auf dem Plus4 und dem auf 64 KByte erweiterten C16/116 bewegte 3-D-Grafiken erzeugt werden. Diese Grafiken sind nicht nur zum Ansehen gedacht, sondern können auch als interessante Vorspänne oder Zwischentitel in eigene Programme eingebunden werden.

Aller guten Dinge sind drei, und so besteht Grafix aus drei Teilprogrammen. Das erste, GRAFIX.START, schaltet den Grafikmodus ein und lädt die zwei restlichen Teile nach. GRAFIX.MC ist der Maschinensprachteil, der für die schnellen Vektoroperationen zuständig ist. Das BASIC-Programm GRAFIX enthält die zur Dateneingabe und Datenkorrektur erforderlichen Routinen. Grafiken können editiert, gespeichert, geladen und überarbeitet werden.

DIE ARBEIT MIT DEM PROGRAMM

Es lassen sich maximal 127 Punkte und 127 Linien definieren. Den einzelnen Koordinaten, Tiefe, Höhe und Breite, können ganzzahlige Werte zwischen −511 und +511 zugewiesen werden. Wenn Sie bei der Definierung der Linien, die an einem Punkt gezogen werden sollen, zum nächsten Punkt übergehen wollen, drücken Sie bei der erneuten Abfrage einfach RETURN.
Beim Löschen eines Punktes werden sämtliche Linien, die von oder zu diesem Punkt gezogen werden, ebenfalls gelöscht. Außerdem verschieben sich die Punktnummern der nachfolgenden Punkte um jeweils Eins nach unten. Die Daten der Linien erfahren hierbei automatisch eine entsprechende Änderung, so daß dadurch keine Fehler entstehen.

BEDIENUNG DER GRAFIK

Während der 3-D-Berechnungen stehen Ihnen folgende Funktionen auf Tastendruck zur Verfügung:
R = Zurück ins Menü.
F = Bild einfrieren.
S = Programm starten.
C = Programm nach Unterbrechung fortsetzen.
L = Der Bildschirm wird von nun an ständig gelöscht. Die Berechnungen werden dadurch schneller. Nachteilig macht sich aber ein Flackern bemerkbar (L-Modus).
T = Die Bildschirmdarstellung erfolgt jetzt ohne Flakkern, die Berechnungen dauern aber länger (T-Modus).

Drehbewegungen:
Cursor rechts = Rechtsdrehung
Cursor links = Linksdrehung
Cursor oben = Drehung nach oben
Cursor unten = Drehung nach unten

Drehbewegungen mit doppelter Schrittweite:
> = Rechtsdrehung + = Drehung nach oben
< = Linksdrehung − = Drehung nach unten
Mit den Tasten 1 bis 8 werden beide Winkel gleichzeitig beeinflußt.

Weitere Bewegungen:
O = Drehwinkel auf Null
G = Objekt näher heranholen
K = Objekt weiter wegrücken
N = Drehen und heranholen
M = Drehen und wegrücken

Neben Laden, Speichern, Ändern, Löschen und Hinzufügen von Daten bietet das Menü auch die Möglichkeit, Bewegungsabläufe vorzuprogrammieren. In diesem Menü-Unterpunkt ist die Taste mit der entsprechenden Bewegungsfunktion zu drücken und danach anzugeben, wie viele solcher Bewegungen, maximal 255, das Objekt ausführen soll.

Eingabebeispiel:

Anzahl der Punkte? 5

Punkt	Tiefe	Höhe	Breite
1	0	60	0
2	40	0	40
3	−40	0	40
4	−40	0	−40
5	40	0	−40

Linien von

1 zu 2	2 zu 3
1 zu 3	2 zu 5
1 zu 4	3 zu 4
1 zu 5	4 zu 5

BESONDERE HINWEISE

Die neue Draw-Routine arbeitet teilweise ungenau, wodurch es zu Veränderungen im Farbspeicher kom-

men kann. Dieser Fehler tritt allerdings nur im L-Modus auf.
Wenn ein Objekt sehr nahe an den Betrachter herangerückt wurde, wird, um Fehlberechnungen vorzubeugen, die Abbildung des Punktes auf 160,100 gesetzt. Das Programm ist mit einem vollständigen Clipp-Teil versehen, so daß auch sehr große Objekte dargestellt werden können.

EINGABE UND ABSPEICHERN DES GRAFIX-PROGRAMMES

Da die Datasette nur der Reihe nach zu lesen vermag, müssen Datasetten-Benutzer sich beim Abspeichern der Programmteile an eine gewisse Reihenfolge halten:

1. GRAFIX.START
2. GRAFIX.MC
3. GRAFIX
4. GRAFIX.CBM

GRAFIX.START und GRAFIX sind reine BASIC-Programme, so daß hierbei keine Besonderheiten zu beachten sind.
GRAFIX.MC und GRAFIX.CBM sind gemäß den im Programmvorspann vermerkten Anweisungen mit Hilfe des Maschinenmonitors TEDMON abzusaven. Der Maschinenteil liegt bei $5750 bis $6E00, die Daten der Objekte bei $1000 bis $1800 im Speicher.
GRAFIX.CBM ist ein Datenfile, das das Commodore-Emblem in 3-D enthält. Sie können nach dem Start von GRAFIX.START diese Grafik durch entsprechende Wahl einladen. GRAFIX verwendet im T-Modus noch den Bereich $8000 bis $9FFF als Pseudo-Grafikspeicher.

Lesen Sie bitte weiter auf Seite 62

```
10 rem grafix.start=============p4  <mo>
20 rem (p) commodore welt team      <ho>
30 rem ========================     <ng>
40 rem (c)  by frank koster         <ge>
50 rem                              <pd>
60 rem                              <ah>
70 rem basic v3.5 und m-code        <jd>
80 rem plus4 (c16/116 +64 kb)       <fd>
90 rem ========================     <jg>
100 graphic1,1:graphic0             <oa>
110 x$=right$(str$(peek(174)),1)    <fd>
120 a$="grafix":r$=chr$(13)         <em>
130 d$=chr$(34)                     <op>
140 b$=d$+a$+".mc"+d$+","+x$+",1"+
r$                                  <oj>
150 c$=d$+a$+d$+","+x$+r$           <nn>
160 key1,"lO"+b$+"lO"+c$+"rU"+r$    <nf>
170 poke2035,0:sys56364:end         <df>
180 rem ========================    <cn>
190 rem dieses programm laedt       <cc>
200 rem "GRAFIX.MC" und "GRAFIX"    <hj>
210 rem nach und startet das        <af>
220 rem letztere.                   <pm>
230 rem ========================    <jc>
240 rem  p r o g r a m m e n d e    <nf>
250 rem ========================    <jf>
```

```
GRAFIX.MC=======================p4
(p) COMMODORE WELT TEAM
==================================
(c) by Frank Koster
Plus4 (C16/116 + 64 KB)
==================================
Das Programm ist mit dem Maschinen-
sprachmonitor TEDMON unter Zuhilfe-
nahme des Pruefsummenprogrammes
CHECKMON einzugeben und auf Disket-
te mit s"grafix.mc",8,5750,6e00 ab-
zuspeichern, auf Kassette mit
s"grafix.mc",1,5750,6e00.
==================================
>5750 a5 d4 c5 da 90 49 d0 2f :<53>
>5758 a5 d3 c5 d9 90 41 d0 27 :<e4>
>5760 20 da 58 38 a5 d0 e5 d6 :<be>
>5768 b0 10 49 ff 69 02 aa ea :<4a>
>5770 ea ca d0 01 60 20 ea 59 :<11>
>5778 d0 f7 aa e8 ea ea ca d0 :<45>
>5780 01 60 20 4a 59 d0 f7 a5 :<84>
>5788 d0 a4 d6 85 d6 84 d0 a5 :<31>
>5790 d3 a4 d9 85 d9 84 d3 a5 :<6f>
>5798 d4 a4 da 85 da 84 d4 20 :<6a>
>57a0 da 58 38 a5 d9 e5 d3 85 :<ce>
>57a8 d3 85 dc a5 da e5 d4 85 :<33>
>57b0 d4 85 dd 38 a5 d0 e5 d6 :<35>
>57b8 90 1a d0 21 a0 00 e6 dc :<46>
>57c0 d0 02 e6 dd e6 dd c6 dc :<c4>
>57c8 d0 05 c6 dd d0 01 60 20 :<d8>
>57d0 bc 59 d0 f2 49 ff a8 c8 :<78>
>57d8 84 d0 4c 5c 58 85 d0 e6 :<b5>
>57e0 d0 e6 dc e6 d3 d0 02 e6 :<ed>
>57e8 d4 e6 dd 20 60 5a a0 00 :<65>
>57f0 a5 d6 85 e0 a5 d7 85 e1 :<62>
>57f8 aa a5 d8 85 e2 8a 29 0f :<e8>
>5800 85 e1 8a 46 e2 6a 46 e2 :<c1>
>5808 6a 46 e2 6a 46 e2 6a f0 :<6d>
>5810 09 aa ca f0 05 20 4a 59 :<90>
>5818 d0 f8 c6 dc d0 05 c6 dd :<3c>
>5820 d0 01 60 18 a5 e0 65 d6 :<33>
>5828 85 e0 a5 e1 65 d7 85 e1 :<8a>
>5830 aa a5 e2 65 d8 85 e2 8a :<2a>
>5838 29 0f 85 e1 8a 46 e2 6a :<38>
>5840 46 e2 6a 46 e2 6a 46 e2 :<84>
>5848 6a d0 05 20 bc 59 d0 ca :<c7>
>5850 aa 20 7f 59 ca f0 c3 20 :<c3>
>5858 4a 59 d0 f8 e6 d0 e6 dc :<47>
>5860 e6 d3 d0 02 e6 d4 e6 dd :<d5>
>5868 20 60 5a a0 00 a5 d6 85 :<5a>
>5870 e0 a5 d7 85 e1 aa a5 d8 :<43>
>5878 85 e2 8a 29 0f 85 e1 8a :<9b>
>5880 46 e2 6a 46 e2 6a 46 e2 :<04>
>5888 6a 46 e2 6a f0 09 aa ca :<ec>
>5890 f0 05 20 ea 59 d0 f8 c6 :<b3>
>5898 dc d0 05 c6 dd d0 01 60 :<85>
>58a0 18 a5 e0 65 d6 85 e0 a5 :<7e>
```

GRAFIX-BEWEGUNG IN 3D - 2 -

```
>58a8 e1 65 d7 85 e1 aa a5 e2 :<52>
>58b0 65 d8 85 e2 8a 29 0f 85 :<07>
>58b8 e1 8a 46 e2 6a 46 e2 6a :<e3>
>58c0 46 e2 6a 46 e2 6a d0 05 :<a1>
>58c8 20 bc 59 d0 ca aa 20 1f :<ae>
>58d0 5a ca f0 c3 20 ea 59 d0 :<3a>
>58d8 f8 ea a5 d0 48 aa 29 07 :<65>
>58e0 85 05 8a 29 f8 0a 26 d1 :<ce>
>58e8 0a 26 d1 0a 26 d1 85 26 :<c3>
>58f0 a6 d1 86 27 0a 26 d1 0a :<6a>
>58f8 26 d1 18 65 26 65 05 85 :<2f>
>5900 26 90 02 e6 27 a5 d1 65 :<85>
>5908 27 69 80 85 27 a5 d3 a8 :<20>
>5910 29 07 aa 98 29 f8 18 65 :<e1>
>5918 26 85 26 a5 27 65 d4 85 :<47>
>5920 27 a9 00 e8 38 6a ca d0 :<9e>
>5928 fc 85 28 a9 08 e5 05 85 :<b7>
>5930 06 a0 00 78 8d 3f ff b1 :<8d>
>5938 26 05 28 91 26 8d 3e ff :<21>
>5940 58 ea e6 05 e6 06 68 85 :<40>
>5948 d0 60 c6 05 f0 11 c6 26 :<6f>
>5950 78 8d 3f ff b1 26 05 28 :<34>
>5958 91 26 8d 3e ff 58 60 a9 :<d2>
>5960 08 85 05 38 a5 26 e9 39 :<ef>
>5968 85 26 b0 02 c6 27 c6 27 :<1f>
>5970 78 8d 3f ff b1 26 05 28 :<f4>
>5978 91 26 8d 3e ff 58 60 c6 :<fc>
>5980 05 f0 15 c6 26 46 28 f0 :<ce>
>5988 46 78 8d 3f ff b1 26 05 :<df>
>5990 28 91 26 8d 3e ff 58 60 :<1b>
>5998 a9 08 85 05 38 a5 26 e9 :<e8>
>59a0 39 85 26 b0 02 c6 27 c6 :<c2>
>59a8 27 46 28 f0 22 78 8d 3f :<b2>
>59b0 ff b1 26 05 28 91 26 8d :<60>
>59b8 3e ff 58 60 46 28 f0 0f :<4c>
>59c0 78 8d 3f ff b1 26 05 28 :<8a>
>59c8 91 26 8d 3e ff 58 60 66 :<92>
>59d0 28 a5 26 69 08 85 26 90 :<ca>
>59d8 02 e6 27 78 8d 3f ff b1 :<81>
>59e0 26 05 28 91 26 8d 3e ff :<92>
>59e8 58 60 c6 06 f0 11 e6 26 :<33>
>59f0 78 8d 3f ff b1 26 05 28 :<f3>
>59f8 91 26 8d 3e ff 58 60 a9 :<91>
>5a00 08 85 06 18 a5 26 69 39 :<0a>
>5a08 85 26 90 02 e6 27 e6 27 :<da>
>5a10 78 8d 3f ff b1 26 05 28 :<b3>
>5a18 91 26 8d 3e ff 58 60 c6 :<c2>
>5a20 06 f0 15 e6 26 46 28 f0 :<11>
>5a28 a6 78 8d 3f ff b1 26 05 :<7f>
>5a30 28 91 26 8d 3e ff 58 60 :<ca>
>5a38 a9 08 85 06 18 a5 26 69 :<fa>
>5a40 39 85 26 90 02 e6 27 e6 :<cb>
>5a48 27 46 28 f0 82 78 8d 3f :<68>
>5a50 ff b1 26 05 28 91 26 8d :<9e>
>5a58 3e ff 58 60 ea ea ea ea :<6e>
>5a60 a0 00 84 86 84 87 84 88 :<7e>
>5a68 84 d6 84 d7 84 d8 a5 d0 :<b7>
>5a70 84 d0 a2 00 86 d2 0a 26 :<0c>
>5a78 d2 0a 26 d2 0a 26 d2 0a :<3a>
>5a80 26 d2 85 d1 e6 87 a5 d3 :<e5>
>5a88 84 d3 a6 d4 86 d5 d0 09 :<57>
>5a90 06 87 26 88 0a 26 d5 f0 :<f7>
>5a98 f7 0a 26 d5 06 87 26 88 :<d4>
>5aa0 0a 26 d5 06 87 26 88 0a :<99>
>5aa8 26 d5 06 87 26 88 85 d4 :<6e>
>5ab0 38 a5 d1 e5 d4 a8 a5 d2 :<f5>
>5ab8 e5 d5 90 0c 84 d1 85 d2 :<59>
>5ac0 a5 88 85 d8 a5 87 85 d7 :<6a>
>5ac8 46 d5 66 d4 66 d3 46 88 :<fb>
>5ad0 66 87 66 86 90 03 60 3d :<d8>
>5ad8 47 38 a5 d0 e5 d3 aa a5 :<c1>
>5ae0 d1 e5 d4 a8 a5 d2 e5 d5 :<ac>
>5ae8 90 de 86 d0 84 d1 85 d2 :<a8>
>5af0 18 a5 d6 65 86 85 d6 a5 :<01>
>5af8 d7 65 87 85 d7 a5 d8 65 :<d5>
>5b00 88 85 d8 4c c8 5a ea ea :<e7>
>5b08 a9 80 85 9a a9 01 a6 d0 :<34>
>5b10 a0 00 ad 5e 06 20 ba ff :<0f>
>5b18 ad 5e 06 a2 5f a0 06 20 :<95>
>5b20 bd ff a2 00 a0 10 86 d8 :<6a>
>5b28 84 d9 a9 d8 a0 18 20 d8 :<23>
>5b30 ff a9 00 85 9a 60 00 00 :<bd>
>5b38 00 00 00 00 00 00 00 00 :<cb>
>5b40 00 00 00 00 78 8d 3f ff :<54>
>5b48 b1 26 05 28 91 26 8d 3e :<1e>
>5b50 ff 58 00 00 a2 00 a5 88 :<b2>
>5b58 f0 22 a5 87 c9 0f 90 0f :<22>
>5b60 a9 68 e5 87 a8 a9 80 85 :<dd>
>5b68 8b b9 f7 03 85 89 60 69 :<41>
>5b70 4b a8 a9 80 85 8b b9 f7 :<e7>
>5b78 03 85 89 60 a5 87 c9 b4 :<a7>
>5b80 90 0d e9 b4 a8 a9 80 85 :<49>
>5b88 8b b9 f7 03 85 89 60 c9 :<54>
>5b90 5b 90 0d a9 b4 e5 87 a8 :<05>
>5b98 86 8b b9 f7 03 85 89 60 :<d7>
>5ba0 a8 86 8b b9 f7 03 85 89 :<fb>
>5ba8 60 a2 00 a5 88 f0 20 a5 :<96>
>5bb0 87 c9 0f 90 0b e9 0e a8 :<bb>
>5bb8 b9 f7 03 85 89 86 8b 60 :<c8>
>5bc0 a9 0f e5 87 a8 a9 80 85 :<6d>
>5bc8 8b b9 f7 03 85 89 60 a5 :<62>
>5bd0 87 c9 b5 90 0f 49 ff 69 :<7c>
>5bd8 0e a8 a9 80 85 8b b9 f7 :<9d>
>5be0 03 85 89 60 c9 5a 90 0d :<9b>
>5be8 e9 5a a8 a9 80 85 8b b9 :<97>
>5bf0 f7 03 85 89 60 a9 5b e5 :<cf>
>5bf8 87 a8 86 8b b9 f7 03 85 :<04>
>5c00 89 60 06 8c 26 8d 26 10 :<b8>
>5c08 06 8c 26 8d 26 10 a6 8c :<1a>
>5c10 a5 8d 85 8c a5 10 85 8d :<a9>
>5c18 a9 00 85 10 8a 10 06 e6 :<2b>
>5c20 8c d0 02 e6 8d 60 a5 c6 :<70>
>5c28 c9 40 b0 03 8d cf 07 ad :<a7>
>5c30 cf 07 c9 30 d0 28 ad d2 :<58>
>5c38 07 d0 0e ad d3 07 d0 17 :<a6>
>5c40 a9 66 8d d3 07 ee d2 07 :<43>
```

```
>5c48 60 ad d3 07 d0 09 ce d2 :<a4>
>5c50 07 a9 fe 8d d3 07 60 ce :<c8>
>5c58 d3 07 ce d3 07 60 c9 33 :<e3>
>5c60 d0 21 ad d2 07 f0 10 ad :<b8>
>5c68 d3 07 c9 66 90 09 a9 00 :<a7>
>5c70 8d d2 07 8d d3 07 60 ee :<d2>
>5c78 d3 07 ee d3 07 d0 03 ee :<bd>
>5c80 d2 07 60 c9 2b d0 28 ad :<12>
>5c88 d0 07 d0 0e ad d1 07 d0 :<9d>
>5c90 17 a9 66 8d d1 07 ee d0 :<41>
>5c98 07 60 ad d1 07 d0 09 ce :<a6>
>5ca0 d0 07 a9 fe 8d d1 07 60 :<e3>
>5ca8 ce d1 07 ce d1 07 60 c9 :<1a>
>5cb0 28 d0 21 ad d0 07 f0 10 :<61>
>5cb8 ad d1 07 c9 66 90 09 a9 :<8d>
>5cc0 00 8d d0 07 8d d1 07 60 :<c2>
>5cc8 ee d1 07 ee d1 07 d0 03 :<6c>
>5cd0 ee d0 07 60 4c 68 6d ea :<84>
>5cd8 ea ea 60 a5 88 45 8e 85 :<f9>
>5ce0 8f a9 00 85 88 85 26 85 :<9f>
>5ce8 27 85 28 a6 8d a4 8c 98 :<af>
>5cf0 29 01 f0 0c a5 26 65 86 :<b3>
>5cf8 85 26 a5 27 65 87 85 27 :<e6>
>5d00 06 86 26 87 98 29 02 f0 :<52>
>5d08 0c a5 26 65 86 85 26 a5 :<2b>
>5d10 27 65 87 85 27 06 86 26 :<18>
>5d18 87 98 29 04 f0 0c a5 26 :<c5>
>5d20 65 86 85 26 a5 27 65 87 :<9d>
>5d28 85 27 06 86 26 87 98 29 :<d7>
>5d30 08 f0 0c a5 26 65 86 85 :<e5>
>5d38 26 a5 27 65 87 85 27 06 :<ca>
>5d40 86 26 87 98 29 10 f0 0c :<01>
>5d48 a5 26 65 86 85 26 a5 27 :<26>
>5d50 65 87 85 27 06 86 26 87 :<bd>
>5d58 98 29 20 f0 0c a5 26 65 :<68>
>5d60 86 85 26 a5 27 65 87 85 :<e7>
>5d68 27 06 86 26 87 26 88 98 :<f6>
>5d70 29 40 f0 12 a5 26 65 86 :<62>
>5d78 85 26 a5 27 65 87 85 27 :<e6>
>5d80 a5 28 65 88 85 28 06 86 :<c5>
>5d88 26 87 26 88 98 29 80 f0 :<ec>
>5d90 12 a5 26 65 86 85 26 a5 :<62>
>5d98 27 65 87 85 27 a5 28 65 :<9a>
>5da0 88 85 28 06 86 26 87 26 :<d2>
>5da8 88 8a 29 01 f0 12 a5 26 :<b4>
>5db0 65 86 85 26 a5 27 65 87 :<2d>
>5db8 85 27 a5 28 65 88 85 28 :<16>
>5dc0 06 86 26 87 26 88 8a 29 :<0f>
>5dc8 02 f0 0c a5 27 65 87 85 :<21>
>5dd0 27 a5 28 65 88 85 28 06 :<ee>
>5dd8 26 26 27 26 28 a6 26 a5 :<f1>
>5de0 27 85 26 a5 28 85 27 a5 :<bb>
>5de8 8f 85 28 8a 10 06 e6 26 :<41>
>5df0 d0 02 e6 27 60 00 b1 26 :<58>
>5df8 05 28 91 26 60 ea ea ea :<f9>
>5e00 ea ea ea ea 00 a5 8d d0 :<13>
>5e08 0d a5 8c c9 70 b0 07 a9 :<66>
>5e10 00 85 86 85 87 60 a0 00 :<36>
>5e18 84 8a 84 8b e6 8a a5 8d :<49>
>5e20 06 8a 26 8b 06 8c 2a 10 :<11>
>5e28 f7 c9 81 b0 11 a8 a5 8c :<30>
>5e30 d0 0b a5 8a 0a 85 86 a5 :<ac>
>5e38 8b 2a 85 87 60 98 85 8d :<93>
>5e40 49 ff 85 89 a5 8c 49 ff :<1d>
>5e48 a8 c8 84 88 d0 02 e6 89 :<75>
>5e50 a5 8a 85 86 a5 8b 85 87 :<65>
>5e58 46 8d 66 8c 46 8b 66 8a :<0d>
>5e60 90 03 4c 87 5e 38 a5 88 :<7b>
>5e68 e5 8c a8 a5 89 e5 8d 90 :<90>
>5e70 e7 85 89 84 88 18 a5 86 :<16>
>5e78 65 8a 85 86 a5 87 65 8b :<ec>
>5e80 85 87 4c 58 5e ea ea a4 :<4e>
>5e88 89 c4 8d 90 0e d0 06 a4 :<7e>
>5e90 88 c4 8c 90 06 e6 86 d0 :<a0>
>5e98 02 e6 87 60 00 00 00 00 :<72>
>5ea0 00 00 00 00 00 00 00 00 :<9f>
>5ea8 00 00 00 46 27 66 26 46 :<ad>
>5eb0 8d 66 8c a5 28 45 8e d0 :<e1>
>5eb8 12 18 a5 26 65 8c 85 8c :<c4>
>5ec0 a5 27 65 8d 85 8d a5 28 :<e0>
>5ec8 85 8e 60 a5 8e 85 8f a5 :<08>
>5ed0 28 30 28 85 8e 38 a5 26 :<1c>
>5ed8 e5 8c 85 8c a5 27 e5 8d :<ba>
>5ee0 85 8d b0 16 49 ff 85 8d :<8d>
>5ee8 a5 8c 49 ff 69 01 d0 02 :<b4>
>5ef0 e6 8d 85 8c a5 8e 49 80 :<0d>
>5ef8 85 8e 60 a5 8f 85 8e 38 :<b7>
>5f00 a5 8c e5 26 85 8c a5 8d :<19>
>5f08 e5 27 85 8d b0 14 49 ff :<38>
>5f10 85 8d a5 8c 49 ff 69 01 :<1b>
>5f18 d0 02 e6 8d 85 8c a5 28 :<76>
>5f20 85 8e 60 00 00 00 00 00 :<bd>
>5f28 00 00 00 00 00 00 00 00 :<af>
>5f30 00 00 46 27 66 26 46 8d :<01>
>5f38 66 8c a5 28 45 8e f0 03 :<b8>
>5f40 4c b9 5e ea ea a5 28 4c :<48>
>5f48 d3 5e a9 00 85 8c 85 8d :<45>
>5f50 a5 88 45 8b 85 8e a6 89 :<2d>
>5f58 18 8a 29 01 f0 0c a5 8c :<06>
>5f60 65 86 85 8c a5 8d 65 87 :<04>
>5f68 85 8d 06 86 26 87 8a 29 :<a6>
>5f70 02 f0 0c a5 8c 65 86 85 :<c7>
>5f78 8c a5 8d 65 87 85 8d 06 :<eb>
>5f80 86 26 87 8a 29 04 f0 0c :<b0>
>5f88 a5 8c 65 86 85 8c a5 8d :<2f>
>5f90 65 87 85 8d 06 86 26 87 :<e7>
>5f98 8a 29 08 f0 0c a5 8c 65 :<57>
>5fa0 86 85 8c a5 8d 65 87 85 :<74>
>5fa8 8d 06 86 26 87 8a 29 10 :<70>
>5fb0 f0 0c a5 8c 65 86 85 8c :<04>
>5fb8 a5 8d 65 87 85 8d 06 86 :<84>
>5fc0 26 87 8a 29 20 f0 0c a5 :<d8>
>5fc8 8c 65 86 85 8c a5 8d 65 :<4a>
>5fd0 87 85 8d 06 86 26 87 8a :<3c>
>5fd8 29 40 f0 0c a5 8c 65 86 :<52>
>5fe0 85 8c a5 8d 65 87 85 8d :<e4>
```

```
>5fe8 06 86 26 87 8a 29 80 f0 :<a5>
>5ff0 0c a5 8c 65 86 85 8c a5 :<57>
>5ff8 8d 65 87 85 8d 60 ea ad :<8b>
>6000 00 10 85 42 20 26 5c ad :<98>
>6008 d3 07 85 87 ad d2 07 85 :<ac>
>6010 88 20 54 5b a5 89 85 03 :<db>
>6018 a5 8b 85 04 20 a9 5b a5 :<31>
>6020 89 85 05 a5 8b 85 06 ad :<05>
>6028 d0 07 85 88 ad d1 07 85 :<4e>
>6030 87 20 54 5b a5 89 85 e0 :<28>
>6038 a5 8b 85 e1 20 a9 5b a5 :<da>
>6040 89 85 e2 a5 8b 85 e3 a0 :<09>
>6048 00 84 2a a4 2a b9 01 10 :<c1>
>6050 85 d0 85 86 b9 02 10 aa :<8c>
>6058 29 7f 85 d1 85 87 8a 29 :<08>
>6060 80 85 d2 85 88 b9 01 11 :<89>
>6068 85 d3 b9 02 11 aa 29 7f :<92>
>6070 85 d4 8a 29 80 85 d5 b9 :<06>
>6078 01 12 85 d6 b9 02 12 aa :<5f>
>6080 29 7f 85 d7 8a 29 80 85 :<3a>
>6088 d8 84 2a a5 05 85 89 a5 :<d5>
>6090 06 85 8b 20 4a 5f a5 8c :<ed>
>6098 85 26 a5 8d 85 27 a5 8e :<07>
>60a0 85 28 a5 d6 85 86 a5 d7 :<96>
>60a8 85 87 a5 d8 85 88 a5 03 :<7e>
>60b0 85 89 a5 04 85 8b 20 4a :<e5>
>60b8 5f 20 ab 5e 20 02 5c a5 :<b6>
>60c0 8c 85 86 85 d9 a5 8d 85 :<64>
>60c8 87 85 da a5 8e 85 88 85 :<ae>
>60d0 db a5 e2 85 89 a5 e3 85 :<c1>
>60d8 8b 20 4a 5f 20 08 5c 18 :<1e>
>60e0 a9 00 85 28 69 01 85 27 :<8c>
>60e8 a9 35 85 26 20 b3 5e a5 :<ce>
>60f0 8c 85 26 a5 8d 85 27 a5 :<07>
>60f8 8e 85 28 a5 d3 85 86 a5 :<41>
>6100 d4 85 87 a5 d5 85 88 a5 :<95>
>6108 e0 85 89 a5 e1 85 8b 20 :<da>
>6110 4a 5f 20 08 5c 20 b3 5e :<ce>
>6118 20 05 5e a5 86 85 dc a5 :<0c>
>6120 87 85 dd a5 8e 85 de a5 :<a0>
>6128 d6 85 86 a5 d7 85 87 a5 :<22>
>6130 d8 85 88 a5 05 85 89 a5 :<81>
>6138 06 85 8b 20 4a 5f a5 8c :<7d>
>6140 85 26 a5 8d 85 27 a5 8e :<d7>
>6148 85 28 a5 d0 85 86 a5 d1 :<91>
>6150 85 87 a5 d2 85 88 a5 03 :<fa>
>6158 85 89 a5 04 85 8b 20 4a :<22>
>6160 5f 20 32 5f 20 02 5c a5 :<68>
>6168 dc 85 86 a5 dd 85 87 a5 :<08>
>6170 de 85 88 20 db 5c a9 00 :<ed>
>6178 85 8d 85 8e a9 a0 85 8c :<13>
>6180 20 b3 5e a4 2a a5 8c 99 :<ec>
>6188 00 08 a5 8d 05 8e 99 01 :<2a>
>6190 08 84 2a a5 d3 85 86 a5 :<93>
>6198 d4 85 87 a5 d5 85 88 a5 :<63>
>61a0 e2 85 89 a5 e3 85 8b 20 :<4a>
>61a8 4a 5f a5 8c 85 26 a5 8d :<bb>
>61b0 85 27 a5 8e 85 28 a5 d9 :<77>
>61b8 85 86 a5 da 85 87 a5 db :<d6>
>61c0 85 88 a5 e0 85 89 a5 e1 :<3f>
>61c8 85 8b 20 4a 5f 20 32 5f :<15>
>61d0 20 02 5c a5 dc 85 86 a5 :<7f>
>61d8 dd 85 87 a5 de 85 88 20 :<bf>
>61e0 db 5c a9 00 85 8d 85 8e :<be>
>61e8 a9 64 85 8c 20 b3 5e a4 :<d8>
>61f0 2a a5 8c 99 00 09 a5 8d :<83>
>61f8 05 8e 99 01 09 c8 c8 84 :<3c>
>6200 2a c6 42 f0 03 4c 4b 60 :<dc>
>6208 ad 0a 59 10 4c 20 5d 6b :<46>
>6210 86 de a6 de bc 02 13 d0 :<72>
>6218 03 4c d0 6b 88 98 0a a8 :<25>
>6220 b9 00 08 85 d3 b9 01 08 :<50>
>6228 85 d4 b9 00 09 85 d0 b9 :<57>
>6230 01 09 85 d1 bc 01 13 e8 :<05>
>6238 e8 86 de 88 98 0a a8 b9 :<7e>
>6240 00 08 85 d9 b9 01 08 85 :<bf>
>6248 da b9 00 09 85 d6 b9 01 :<58>
>6250 09 85 d7 20 b0 66 4c 12 :<36>
>6258 62 20 ff 6c 86 de a6 de :<c8>
>6260 bc 02 13 d0 05 84 10 4c :<b7>
>6268 ff 5f 88 98 0a a8 b9 00 :<ae>
>6270 08 85 d3 b9 01 08 85 d4 :<84>
>6278 b9 00 09 85 d0 b9 01 09 :<34>
>6280 85 d1 bc 01 13 e8 e8 86 :<8c>
>6288 de 88 98 0a a8 b9 00 08 :<a8>
>6290 85 d9 b9 01 08 85 da b9 :<09>
>6298 00 09 85 d6 b9 01 09 85 :<17>
>62a0 d7 20 b0 66 4c 5e 62 00 :<b5>
>62a8 00 00 00 00 00 00 00 00 :<b3>
>62b0 00 00 00 00 00 00 00 00 :<c3>
>62b8 00 00 00 c9 11 d0 03 68 :<99>
>62c0 68 60 c9 15 d0 03 4c 26 :<1b>
>62c8 5c c9 23 d0 0f a9 00 8d :<30>
>62d0 d0 07 8d d1 07 8d d2 07 :<d2>
>62d8 8d d3 07 60 c9 38 d0 07 :<77>
>62e0 20 36 5c 20 87 5c 60 c9 :<70>
>62e8 3b d0 07 20 62 5c 20 b3 :<1d>
>62f0 5c 60 c9 08 d0 07 20 36 :<c6>
>62f8 5c 20 b3 5c 60 c9 0b d0 :<67>
>6300 07 20 62 5c 20 87 5c 60 :<cc>
>6308 c9 10 d0 0a 20 36 5c 20 :<3b>
>6310 36 5c 20 87 5c 60 c9 13 :<ca>
>6318 d0 0a 20 62 5c 20 62 5c :<8b>
>6320 20 b3 5c 60 c9 18 d0 0a :<22>
>6328 20 87 5c 20 87 5c 20 62 :<80>
>6330 5c 60 c9 1b d0 0a 20 b3 :<7e>
>6338 5c 20 b3 5c 20 36 5c 60 :<8d>
>6340 c9 2f d0 07 20 36 5c 20 :<32>
>6348 36 5c 60 c9 2c d0 07 20 :<c6>
>6350 62 5c 20 62 5c 60 c9 2e :<c2>
>6358 d0 07 20 87 5c 20 87 5c :<7f>
>6360 60 c9 36 d0 07 20 b3 5c :<db>
>6368 20 b3 5c 60 c9 25 d0 1d :<18>
>6370 18 ad e9 60 69 30 8d e9 :<67>
>6378 60 90 11 ac e5 60 c8 c0 :<b0>
>6380 50 90 06 a9 f8 8d e9 60 :<4d>
```

```
>6388 88 8c e5 60 60 c9 1a d0 :<49>
>6390 29 ac e5 60 c0 02 b0 13 :<85>
>6398 38 ad e9 60 e9 30 8d e9 :<3d>
>63a0 60 c9 35 b0 05 a9 35 8d :<5f>
>63a8 e9 60 60 38 ad e9 60 e9 :<f2>
>63b0 30 8d e9 60 b0 f4 ce e5 :<15>
>63b8 60 60 c9 24 d0 0a 20 70 :<37>
>63c0 63 20 1a 63 20 1a 63 60 :<d9>
>63c8 c9 27 d0 0a 20 91 63 20 :<08>
>63d0 1a 63 20 1a 63 60 c9 14 :<a2>
>63d8 d0 0f ad 00 06 f0 01 60 :<83>
>63e0 a9 0d 8d cf 07 60 ea ea :<73>
>63e8 ea c9 0d d0 f8 ad 00 06 :<7a>
>63f0 f0 14 a9 00 8d 03 06 8d :<f2>
>63f8 00 06 ad 59 15 8d 01 06 :<06>
>6400 ad 21 16 8d 02 06 ad 01 :<a4>
>6408 06 20 32 5c ce 02 06 f0 :<fc>
>6410 01 60 ee 03 06 ac 03 06 :<cf>
>6418 ea ea b9 59 15 10 09 a9 :<62>
>6420 01 8d 00 06 8d cf 07 60 :<93>
>6428 8d 01 06 b9 21 16 8d 02 :<1a>
>6430 06 60 a5 d0 a0 ff c8 d9 :<e4>
>6438 50 64 f0 06 be 50 64 10 :<e6>
>6440 f5 60 b9 70 64 85 d0 60 :<91>
>6448 00 00 00 00 00 00 00 00 :<f4>
>6450 38 3b 08 0b 10 13 18 1b :<21>
>6458 23 36 2e 2c 2f 33 30 2b :<b0>
>6460 28 24 27 1a 25 15 11 0d :<c1>
>6468 1c 14 16 2a ff 00 ff ff :<51>
>6470 31 32 33 34 35 36 37 38 :<4c>
>6478 30 2b 2d 3e 3c 80 81 82 :<54>
>6480 83 4d 4e 47 4b 46 52 53 :<1d>
>6488 42 43 54 4c 00 00 00 00 :<ed>
>6490 00 00 00 00 00 00 00 00 :<85>
>6498 00 00 00 00 00 00 00 00 :<95>
>64a0 00 00 00 00 00 00 00 00 :<a5>
>64a8 00 00 00 00 00 00 00 00 :<b5>
>64b0 a9 00 85 8c 85 8d 85 8e :<03>
>64b8 a9 01 85 e6 8a 25 e6 f0 :<70>
>64c0 13 18 a5 8c 65 26 85 8c :<1e>
>64c8 a5 8d 65 27 85 8d a5 8e :<76>
>64d0 65 28 85 8e 06 26 26 27 :<79>
>64d8 26 28 06 e6 90 de 26 e6 :<03>
>64e0 a9 04 85 e7 98 25 e6 f0 :<ad>
>64e8 13 18 a5 8c 65 26 85 8c :<c9>
>64f0 a5 8d 65 27 85 8d a5 8e :<e6>
>64f8 65 28 85 8e 06 26 26 27 :<cf>
>6500 26 28 06 e6 c6 e7 d0 dc :<44>
>6508 a5 8d 46 8e 6a 46 8e 6a :<42>
>6510 46 8e 6a 46 8e 6a 90 06 :<11>
>6518 69 00 d0 02 e6 8e 85 8c :<05>
>6520 a5 8e 85 8d a5 12 85 8e :<e9>
>6528 60 a0 00 84 e3 84 e4 84 :<5a>
>6530 e5 84 e0 84 e1 84 e2 a5 :<a8>
>6538 8d 85 8e a5 8c 85 8d 84 :<39>
>6540 8c 06 8d 26 8e 06 8d 26 :<b1>
>6548 8e 06 8d 26 8e 06 8d 26 :<82>
>6550 8e a5 41 85 42 a5 40 85 :<88>
>6558 41 84 40 06 41 26 42 06 :<ec>
>6560 41 26 42 06 41 26 42 a9 :<d5>
>6568 08 85 e4 a5 42 06 e4 26 :<ed>
>6570 e5 06 41 2a 10 f7 85 42 :<a6>
>6578 38 a5 8d e5 41 aa a5 8e :<8b>
>6580 e5 42 90 0c 85 8e 86 8d :<34>
>6588 a5 e4 85 e1 a5 e5 85 e2 :<05>
>6590 46 42 66 41 66 40 46 e5 :<a3>
>6598 66 e4 66 e3 b0 2d 38 a5 :<fa>
>65a0 8c e5 40 aa a5 8d e5 41 :<30>
>65a8 a8 a5 8e e5 42 90 e1 85 :<16>
>65b0 8e 84 8d 86 8c 18 a5 e0 :<86>
>65b8 65 e3 85 e0 a5 e1 65 e4 :<ec>
>65c0 85 e1 a5 e2 65 e5 85 e2 :<14>
>65c8 4c 90 65 a4 8e c4 42 90 :<86>
>65d0 1a d0 0e a4 8d c4 41 90 :<fe>
>65d8 12 d0 06 a4 8c c4 40 90 :<96>
>65e0 0a e6 e0 d0 06 e6 e1 d0 :<37>
>65e8 02 e6 e2 60 a4 d4 a6 d3 :<1e>
>65f0 a5 03 85 26 a5 04 85 27 :<f4>
>65f8 a5 05 85 28 20 b0 64 a5 :<9b>
>6600 d0 85 26 a5 d1 85 27 a5 :<9e>
>6608 d2 85 28 4c b3 5e a5 d0 :<64>
>6610 85 8c a5 d1 85 8d a9 00 :<94>
>6618 85 8e a5 03 85 40 a5 04 :<5e>
>6620 85 41 a5 05 85 42 20 9a :<42>
>6628 6a a5 e2 f0 0d 85 8e a5 :<e8>
>6630 e1 85 8d a5 e0 85 8c ea :<70>
>6638 ea 60 85 28 a5 e1 85 27 :<ee>
>6640 a5 e0 85 26 a5 d5 85 8e :<5f>
>6648 a5 d4 85 8d a5 d3 85 8c :<4d>
>6650 4c b3 5e a9 00 85 28 a9 :<03>
>6658 01 85 27 a9 3f 85 26 a5 :<e4>
>6660 d5 85 8e a5 d4 85 8d a5 :<2b>
>6668 d3 85 8c 20 3a 5f a6 8c :<82>
>6670 a4 8d 4c f0 65 a9 00 85 :<6f>
>6678 28 85 27 a9 c7 85 26 a5 :<52>
>6680 d2 85 8e a5 d1 85 8d a5 :<b8>
>6688 d0 85 8c 20 3a 5f 4c 16 :<90>
>6690 66 a5 e0 85 03 a5 e1 85 :<ba>
>6698 04 a5 e2 85 05 60 a5 10 :<1e>
>66a0 a4 11 85 11 84 10 60 00 :<4a>
>66a8 00 00 00 00 00 00 00 ea :<ac>
>66b0 a0 00 84 10 84 11 a5 d1 :<e6>
>66b8 10 08 a5 10 09 04 85 10 :<45>
>66c0 d0 10 f0 08 a5 10 09 02 :<67>
>66c8 85 10 d0 06 a5 d0 c9 c8 :<66>
>66d0 b0 f2 a5 d4 10 08 a5 10 :<43>
>66d8 09 08 85 10 d0 14 f0 12 :<4b>
>66e0 c9 02 90 08 a5 10 09 01 :<81>
>66e8 85 10 d0 06 a5 d3 c9 40 :<52>
>66f0 b0 f2 a5 d7 10 08 a5 11 :<18>
>66f8 09 04 85 11 d0 10 f0 08 :<01>
>6700 a5 11 09 02 85 11 d0 06 :<b4>
>6708 a5 d6 c9 c8 b0 f2 a5 da :<7a>
>6710 10 08 a5 11 09 08 85 11 :<3e>
>6718 d0 14 f0 12 c9 02 90 08 :<d5>
>6720 a5 11 09 01 85 11 d0 06 :<ec>
```

```
>6728 a5 d9 c9 40 b0 f2 a5 10 :<a1>
>6730 f0 05 25 11 f0 09 60 a5 :<69>
>6738 11 d0 04 4c 50 57 ea a5 :<a7>
>6740 d1 aa 29 7f 85 d1 8a 29 :<06>
>6748 80 85 d2 a5 d4 aa 29 7f :<cb>
>6750 85 d4 85 8d 8a 29 80 85 :<50>
>6758 d5 85 8e a5 d7 aa 29 7f :<21>
>6760 85 d7 8a 29 80 85 d8 a5 :<2e>
>6768 da aa 29 7f 85 da 85 27 :<83>
>6770 8a 29 80 85 db 85 28 a5 :<08>
>6778 d3 85 8c a5 d9 85 26 20 :<54>
>6780 3a 5f a5 8d d0 47 a5 8c :<1c>
>6788 d0 43 a5 10 aa 29 04 f0 :<9a>
>6790 0a a9 00 85 d0 85 d1 85 :<dc>
>6798 d2 f0 0f 8a 29 02 f0 0a :<a6>
>67a0 a9 00 85 d1 85 d2 a9 c7 :<52>
>67a8 85 d0 a5 11 aa 29 04 f0 :<b1>
>67b0 0a a9 00 85 d6 85 d7 85 :<5e>
>67b8 d8 f0 0f 8a 29 02 f0 0a :<ed>
>67c0 a9 00 85 d7 85 d8 a9 c7 :<7f>
>67c8 85 d6 4c 60 57 a5 8e 10 :<ab>
>67d0 33 a5 d5 a4 db 85 db 84 :<60>
>67d8 d5 a5 d4 a4 da 85 da 84 :<9d>
>67e0 d4 a5 d3 a4 d9 85 d9 84 :<78>
>67e8 d3 a5 d2 a4 d8 85 d8 84 :<0f>
>67f0 d2 a5 d1 a4 d7 85 d7 84 :<8b>
>67f8 d1 a5 d0 a4 d6 85 d6 84 :<04>
>6800 d0 20 9e 66 a6 8c a4 8d :<1d>
>6808 e8 d0 01 c8 86 40 84 41 :<d5>
>6810 a9 00 85 42 a5 d2 85 8e :<82>
>6818 a5 d1 85 8d a5 d0 85 8c :<25>
>6820 a5 d8 85 28 a5 d7 85 27 :<2a>
>6828 a5 d6 85 26 20 3a 5f a5 :<c4>
>6830 8d d0 44 a5 8c d0 40 a5 :<4e>
>6838 10 aa 29 08 f0 06 a9 00 :<0a>
>6840 85 d4 85 d3 a5 11 29 01 :<fd>
>6848 f0 08 a9 01 85 da a9 3f :<90>
>6850 85 d9 20 da 58 38 a5 d9 :<10>
>6858 e5 d3 aa a5 da e5 d4 85 :<68>
>6860 d4 a0 00 e6 d4 e8 d0 02 :<9b>
>6868 e6 d4 ca d0 05 c6 d4 d0 :<55>
>6870 01 60 20 bc 59 d0 f3 e6 :<a5>
>6878 8c d0 02 e6 8d a5 8e 85 :<0e>
>6880 12 a9 00 85 8e 20 29 65 :<52>
>6888 20 91 66 a5 12 10 03 4c :<e2>
>6890 86 69 a5 10 29 08 f0 59 :<31>
>6898 20 ec 65 a5 8e 10 2f 20 :<0a>
>68a0 0e 66 a5 8d f0 0b c9 02 :<0f>
>68a8 90 01 60 a5 8c c9 40 b0 :<cc>
>68b0 f9 a9 00 8d 00 0d 8d 01 :<fe>
>68b8 0d 8d 02 0d a5 8c 8d 03 :<21>
>68c0 0d a5 8d 8d 04 0d a5 8e :<ae>
>68c8 8d 05 0d 4c 11 69 a5 8d :<0d>
>68d0 d0 d8 a5 8c c9 c8 b0 d2 :<56>
>68d8 a9 00 8d 03 0d 8d 04 0d :<16>
>68e0 8d 05 0d 8d 01 0d 8d 02 :<2e>
>68e8 0d a5 8c 8d 00 0d 4c 11 :<5f>
>68f0 69 a5 10 29 04 d0 a8 a5 :<13>
>68f8 d0 8d 00 0d a9 00 8d 01 :<13>
>6900 0d 8d 02 0d 8d 05 0d a5 :<00>
>6908 d3 8d 03 0d a5 d4 8d 04 :<c3>
>6910 0d a5 11 29 01 f0 48 20 :<9d>
>6918 53 66 a5 8d d0 21 a5 8c :<48>
>6920 c9 c8 b0 1b 8d 06 0d a9 :<0b>
>6928 00 8d 07 0d 8d 08 0d 8d :<ba>
>6930 0b 0d a9 01 8d 0a 0d a9 :<4b>
>6938 3f 8d 09 0d 4c 55 6a 20 :<8a>
>6940 75 66 a9 00 8d 07 0d 8d :<13>
>6948 08 0d 8d 0b 0d a9 c7 8d :<c8>
>6950 06 0d a5 8c 8d 09 0d a5 :<12>
>6958 8d 8d 0a 0d 4c 55 6a a5 :<ee>
>6960 11 29 02 d0 da a5 d6 8d :<a1>
>6968 06 0d a5 d7 8d 07 0d a5 :<e9>
>6970 d8 8d 08 0d a5 d9 8d 09 :<10>
>6978 0d a5 da 8d 0a 0d a5 db :<9e>
>6980 8d 0b 0d 4c 55 6a a5 10 :<4f>
>6988 29 08 f0 58 20 ec 65 a5 :<59>
>6990 8e 10 01 60 a5 8d d0 1d :<be>
>6998 a5 8c c9 c8 b0 17 8d 00 :<e9>
>69a0 0d a9 00 8d 01 0d 8d 02 :<4b>
>69a8 0d 8d 03 0d 8d 04 0d 8d :<3a>
>69b0 05 0d 4c 04 6a 20 75 66 :<26>
>69b8 a5 8d f0 0b c9 02 90 01 :<24>
>69c0 60 a5 8c c9 40 b0 f9 a5 :<93>
>69c8 8c 8d 03 0d a5 8d 8d 04 :<47>
>69d0 0d a9 c7 8d 00 0d a9 00 :<f3>
>69d8 8d 01 0d 8d 02 0d 8d 05 :<d1>
>69e0 0d 4c 04 6a a5 10 29 02 :<ca>
>69e8 d0 cb a5 d0 8d 00 0d a5 :<17>
>69f0 d3 8d 03 0d a5 d4 8d 04 :<d5>
>69f8 0d a5 d1 8d 01 0d 8d 02 :<58>
>6a00 0d 8d 05 0d a5 11 29 01 :<46>
>6a08 f0 42 20 53 66 a5 8e 30 :<2e>
>6a10 1d a9 00 8d 07 0d 8d 08 :<d8>
>6a18 0d 8d 0b 0d a9 01 8d 0a :<3a>
>6a20 0d a9 3f 8d 09 0d a5 8c :<0b>
>6a28 8d 06 0d 4c 55 6a 20 0e :<9c>
>6a30 66 a9 00 8d 06 0d 8d 07 :<d1>
>6a38 0d 8d 08 0d 8d 0b 0d a5 :<59>
>6a40 8d 8d 0a 0d a5 8c 8d 09 :<95>
>6a48 0d 4c 55 6a a5 11 29 04 :<bd>
>6a50 d0 dc 4c 65 69 ad 00 0d :<9c>
>6a58 85 d0 ad 01 0d 85 d1 ad :<86>
>6a60 02 0d 85 d2 ad 03 0d 85 :<28>
>6a68 d3 ad 04 0d 85 d4 ad 05 :<5f>
>6a70 0d 85 d5 ad 06 0d 85 d6 :<2f>
>6a78 ad 07 0d 85 d7 ad 08 0d :<19>
>6a80 85 d8 ad 09 0d 85 d9 ad :<c9>
>6a88 0a 0d 85 da ad 0b 0d 85 :<01>
>6a90 db a9 00 85 12 4c 50 57 :<06>
>6a98 ea ea a0 00 84 e3 84 e4 :<7f>
>6aa0 84 e5 84 e0 84 e1 84 e2 :<70>
>6aa8 a5 8d 85 8e a5 8c 85 8d :<8b>
>6ab0 84 8c 06 8d 26 8e 06 8d :<c1>
>6ab8 26 8e 06 8d 26 8e 06 8d :<97>
>6ac0 26 8e a9 01 85 e3 a5 41 :<17>
```

```
>6ac8 06 e3 26 e4 26 e5 06 40 :<9f>
>6ad0 2a 26 42 10 f3 85 41 38 :<33>
>6ad8 a5 8d e5 41 aa a5 8e e5 :<bc>
>6ae0 42 90 10 85 8e 86 8d a5 :<a8>
>6ae8 e3 85 e0 a5 e4 85 e1 a5 :<45>
>6af0 e5 85 e2 46 42 66 41 66 :<1f>
>6af8 40 46 e5 66 e4 66 e3 b0 :<80>
>6b00 2d 38 a5 8c e5 40 aa a5 :<1a>
>6b08 8d e5 41 a8 a5 8e e5 42 :<f2>
>6b10 90 e1 85 8e 84 8d 86 8c :<21>
>6b18 18 a5 e0 65 e3 85 e0 a5 :<4d>
>6b20 e1 65 e4 85 e1 a5 e2 65 :<e0>
>6b28 e5 85 e2 4c f3 6a a4 8e :<02>
>6b30 c4 42 90 1a d0 0e a4 8d :<51>
>6b38 c4 41 90 12 d0 06 a4 8c :<61>
>6b40 c4 40 90 0a e6 e0 d0 06 :<32>
>6b48 e6 e1 d0 02 e6 e2 60 90 :<6d>
>6b50 0a e6 e0 d0 06 e6 e1 d0 :<a4>
>6b58 02 e6 e2 60 ea a9 00 aa :<7c>
>6b60 78 8d 3f ff 9d 00 80 9d :<27>
>6b68 00 81 9d 00 82 9d 00 83 :<30>
>6b70 9d 00 84 9d 00 85 9d 00 :<d7>
>6b78 86 9d 00 87 9d 00 88 9d :<48>
>6b80 00 89 9d 00 8a 9d 00 8b :<73>
>6b88 9d 00 8c 9d 00 8d 9d 00 :<05>
>6b90 8e 9d 00 8f 9d 00 90 9d :<8c>
>6b98 00 91 9d 00 92 9d 00 93 :<c6>
>6ba0 9d 00 94 9d 00 95 9d 00 :<36>
>6ba8 96 9d 00 97 9d 00 98 9d :<97>
>6bb0 00 99 9d 00 9a 9d 00 9b :<ea>
>6bb8 9d 00 9c 9d 00 9d 9d 00 :<ac>
>6bc0 9e 9d 00 9f 8d 3e ff 58 :<ad>
>6bc8 e8 d0 95 60 ea ea ea ea :<25>
>6bd0 a0 00 78 8d 3f ff b9 00 :<30>
>6bd8 80 99 00 20 b9 00 81 99 :<46>
>6be0 00 21 b9 00 82 99 00 22 :<a1>
>6be8 b9 00 83 99 00 23 b9 00 :<56>
>6bf0 84 99 00 24 8d 3e ff 58 :<9f>
>6bf8 c8 d0 d7 ea ea ea ea ea :<bb>
>6c00 78 8d 3f ff b9 00 85 99 :<ed>
>6c08 00 25 b9 00 86 99 00 26 :<b7>
>6c10 b9 00 87 99 00 27 b9 00 :<7b>
>6c18 88 99 00 28 b9 00 89 99 :<0b>
>6c20 00 29 8d 3e ff 58 c8 d0 :<f7>
>6c28 d7 ea ea ea ea 78 8d 3f :<ef>
>6c30 ff b9 00 8a 99 00 2a b9 :<0a>
>6c38 00 8b 99 00 2b b9 00 8c :<08>
>6c40 99 00 2c b9 00 8d 99 00 :<b4>
>6c48 2d b9 00 8e 99 00 2e 8d :<6b>
>6c50 3e ff 58 c8 d0 d7 ea ea :<6f>
>6c58 ea ea 78 8d 3f ff b9 00 :<1e>
>6c60 8f 99 00 2f b9 00 90 99 :<a2>
>6c68 00 30 b9 00 91 99 00 31 :<36>
>6c70 b9 00 92 99 00 32 b9 00 :<b0>
>6c78 93 99 00 33 8d 3e ff 58 :<b6>
>6c80 c8 d0 d7 ea ea ea ea 78 :<a3>
>6c88 8d 3f ff b9 00 94 99 00 :<01>
>6c90 34 b9 00 95 99 00 35 b9 :<93>
>6c98 00 96 99 00 36 b9 00 97 :<73>
>6ca0 99 00 37 b9 00 98 99 00 :<02>
>6ca8 38 b9 00 99 99 00 39 8d :<5a>
>6cb0 3e ff 58 c8 d0 d1 ea ea :<6c>
>6cb8 ea ea 78 8d 3f ff b9 00 :<e5>
>6cc0 9a 99 00 3a b9 00 9b 99 :<1b>
>6cc8 00 3b b9 00 9c 99 00 3c :<c5>
>6cd0 b9 00 9d 99 00 3d b9 00 :<fb>
>6cd8 9e 99 00 3e b9 00 9f 99 :<ee>
>6ce0 00 3f 8d 3e ff 58 c8 d0 :<a4>
>6ce8 d1 a9 00 85 10 4c ff 5f :<7d>
>6cf0 00 00 00 00 00 00 00 00 :<4d>
>6cf8 00 00 00 00 00 00 ea a2 :<a8>
>6d00 00 a9 00 9d 00 20 9d 01 :<59>
>6d08 20 9d 02 20 9d 03 20 9d :<70>
>6d10 04 20 9d 05 20 9d 06 20 :<97>
>6d18 9d 07 20 8a 69 08 aa d0 :<3f>
>6d20 e0 ae 05 6d e8 e0 3f f0 :<b2>
>6d28 1b 8e 05 6d 8e 08 6d 8e :<5f>
>6d30 0b 6d 8e 0e 6d 8e 11 6d :<6f>
>6d38 8e 14 6d 8e 17 6d 8e 1a :<45>
>6d40 6d aa f0 bd a2 20 8e 05 :<5f>
>6d48 6d 8e 08 6d 8e 0b 6d 8e :<7e>
>6d50 0e 6d 8e 11 6d 8e 14 6d :<08>
>6d58 8e 17 6d 8e 1a 6d aa a0 :<e1>
>6d60 41 99 ff 3e 88 d0 fa 60 :<b8>
>6d68 c9 16 d0 31 a9 80 8d 0a :<fe>
>6d70 59 a0 0e b9 43 5b 99 32 :<c4>
>6d78 59 99 4f 59 99 6f 59 99 :<e1>
>6d80 88 59 99 ac 59 99 bf 59 :<01>
>6d88 99 da 59 99 ef 59 99 0f :<7a>
>6d90 5a 99 28 5a 99 4c 5a 88 :<8b>
>6d98 d0 d9 60 ea ea c9 2a d0 :<1b>
>6da0 37 a9 20 8d 0a 59 a0 0e :<c0>
>6da8 b9 f5 5d 99 32 59 99 4f :<bc>
>6db0 59 99 6f 59 99 88 59 99 :<0c>
>6db8 ac 59 99 bf 59 99 da 59 :<05>
>6dc0 99 ef 59 99 0f 5a 99 28 :<c6>
>6dc8 5a 99 4c 5a 88 d0 d9 a9 :<99>
>6dd0 ea 8d 39 59 60 ea ea ea :<a1>
>6dd8 4c bb 62 00 00 00 a9 80 :<d5>
>6de0 85 9a a9 01 a6 d0 a0 01 :<5d>
>6de8 20 ba ff ad 5e 06 a2 5f :<f3>
>6df0 a0 06 20 bd ff a9 00 20 :<75>
>6df8 d5 ff a9 00 85 9a 60 00 :<67>
```

====================================

P R O G R A M M E N D E

====================================

```
GRAFIX ========================P4
(p) COMMODORE WELT TEAM
=================================
(c) by Frank Koster

PLUS4 (C16/116 + 64 KB)
=================================
Programmkopf und Programmende
sind aus Platzgruenden wegzulas-
sen.
=================================
100 color0,1:color1,2:color4,1:gos
ub1390                              <dd>
110 graphic1,1:graphic0,1:print"da
ten laden j/n":getkeyt$:ift$="j"th
en740                               <dn>
120 gosub1160                       <lk>
130 gosub970:input"anzahl der punk
te,die sie definieren wollen";n:po
ke4096,n                            <fg>
140 fort=0ton-1:graphic0,1:print"k
oordinaten von punkt ";t+1:input"t
iefe";l                             <kl>
150 input"hoehe";h:input"breite";b
:x=-l:a=q:gosub750:x=-h:a=e:gosub7
50                                  <lf>
160 x=b:a=i:gosub750:next:h=0       <cf>
170 fort=1ton:graphic0,1:a$=""      <mi>
180 print"zu welchen anderen punkt
en sollen von punkt";t;"linien gez
ogen werden"                        <hf>
190 printt;"->":poke202,peek(202)+
4:poke205,peek(205)-1:input" ";a$:
ifa$=""then210                      <aj>
200 pokep+h,t:pokec+h,val(a$):h=h+
2:a$="":goto190                     <fj>
210 next:pokep+h,0:pokec+h,0:gosub
990                                 <ga>
220 graphic0,1:print"sind die date
n korrekt oder sollen aende"        <pe>
230 print"-rungen vorgenommen werd
en j/n":getkeyt$:ift$="n"then530    <mb>
240 gosub1160                       <mj>
250 print"bei den punkten oder bei
 den linien p/l":getkeyt$:ift$="l"
then400                             <mb>
260 graphic0,1:print"koordinaten v
eraendern/punkt loeschen/neuer pun
kt v/l/n"                           <mg>
270 getkeyt$:ift$="l"then380        <kh>
280 ift$="n"then360                 <kh>
290 graphic0,1:input"welcher punkt
";t:t=t-1:a=q:gosub770:l=-x         <lj>
300 a=e:gosub770:h=-x:a=i:gosub770
:b=x                                <kb>
310 print"alte werte(ti,ho,br)";l,
h,b                                 <eg>
320 input"neue werte(ti,ho,br)";l,
h,b:x=-l:a=q:gosub750               <df>
330 x=-h:a=e:gosub750:x=b:a=i:gosu
b750                                <hm>
340 print"weitere punkte korrigier
en j/n"                             <nl>
350 getkeyt$:ift$="j"then260:else2
20                                  <if>
360 t=peek(4096)+1:print"der neue
punkt hat die nummer ";t:poke4096,
t:t=t-1                             <mo>
370 input"tiefe";l:input"hoehe";h:
input"breite";b:x=-l:a=q:gosub750:
goto330                             <bb>
380 graphic0,1:input"welcher punkt
 soll geloescht werden";t:n=peek(4
096)                                <jb>
390 gosub790:h=0:gosub830:h=0:gosu
b1120:goto340                       <on>
400 graphic0,1:print"linien loesch
en/neue linie/a.oder endpunkt vera
endern l/n/v"                       <nb>
410 getkeyt$:ift$="l"then470:elsei
ft$="v"then490                      <ip>
420 graphic0,1:print"welche linie
soll hinzugefuegt werden"           <ih>
430 input"anfangspunkt";x:input"en
dpunkt";y:h=0                       <mb>
440 ifpeek(p+h)<>0then h=h+2:goto4
40                                  <lf>
450 pokep+h,x:pokec+h,y:pokep+2+h,
0:pokec+2+h,0                       <nd>
460 print"weitere linien korrigier
en j/n":getkeyt$:ift$="j"then400:e
lse220                              <bd>
470 input"anfangspunkt der zu loes
chenden linie";x:input"endpunkt";y
:h=0                                <ei>
480 gosub870:goto460                <hf>
490 graphic0,1:print"welche linie
soll veraendert werden:"            <lo>
500 input"alter anfangspunkt";x:in
put"alter endpunkt";y               <dm>
510 input"neuer anfangspunkt";l:in
put"neuer endpunkt";b               <nn>
520 h=0:gosub940:goto460            <pf>
530 ifpeek(1016)=2andpeek(1020)=11
then550                             <eb>
540 fort=0to90:poke1015+t,int(sin(
t*π/180)*128+.5):next               <pa>
550 color0,1:color1,2:graphic1,1:p
oke1999,1:poke1536,1:sys24575       <lk>
560 graphic0,1:fort=1319to1328:pok
et,00:next:gosub1160:gosub1390:gra
phic0,1                             <ml>
570 print"verzeichnis der daten j/
n":getkeyt$:ift$="j"thengosub990    <fh>
580 gosub1160:print"daten aendern
j/n":getkeyt$:ift$="j"then250       <fb>
```

```
590 gosub1160:print"daten abspeich
ern j/n":getkeyt$:ift$="j"then670   <ei>
600 gosub1160:print"zurueck zur gr
afik j/n":getkeyt$:ift$="j"then530  <pa>
610 gosub1160:print"neue daten lad
en j/n":getkeyt$:ift$="j"then740    <dl>
620 print" bewegungsablaeufe def.
j/n":getkeyt$:ift$="j"thengosub117
0:goto640                           <gd>
630 gosub1160:print"  daten des be
weg.ablaufs j/n":getkeyt$:ift$="j"
thengosub1230                       <lg>
640 gosub1160:print"  beweg.ablauf
 aendern":getkeyt$:ift$="j"then 13
10                                  <mi>
650 gosub1160:print"programm neu s
tarten j/n":getkeyt$:ift$="j"then
run                                 <bb>
660 gosub1160:print"3-d programm b
eenden j/n":getkeyt$:ift$="j"then
new:else 570                        <cg>
670 graphic0,1:print"kassette/disk
ette k/d":gosub1160:getkeyt$        <of>
680 ift$="d"thenpoke208,8:elsepoke
208,1:print"nach spulen taste drue
cken":getkeyt$                      <ia>
690 gosub1160:sys23304:getkeyt$:gr
aphic0,1:goto540                    <aj>
700 graphic0,1:print"kassette/disk
ette k/d":fort=0to35:next:getkeyt$  <ed>
710 ift$="d"thenpoke208,8:goto730:
elsepoke208,1                       <po>
720 print" sie koennen jetzt spule
n--danach taste  druecken":fort=0t
o30:next:getkeyt$                   <mj>
730 fort=0to50:next:sys28126:graph
ic0,1:goto540                       <df>
740 gosub970:goto700                <np>
750 ifx<0thenx=-x+32768             <je>
760 hb=int(x/256):lb=x-hb*256:poke
a+2*t,lb:pokea+1+2*t,hb:return      <mn>
770 lb=peek(a+2*t):hb=peek(a+1+2*t
):x=hb*256+lb:ifx>32767thenx=-(x-3
2768)                               <km>
780 return                          <km>
790 forl=tton:pokeq-2+2*l,peek(q+2
*l):pokeq-1+2*l,peek(q+2*l+1)       <ff>
800 pokee-2+2*l,peek(e+2*l):pokee-
1+2*l,peek(e+2*l+1)                 <em>
810 pokei-2+2*l,peek(i+2*l):pokei-
1+2*l,peek(i+2*l+1):next:poke4096,
n-1                                 <ko>
820 return                          <pm>
830 p1=peek(p+h):p2=peek(c+h):ifp1
=t or p2=tthenb=h+2:goto850         <kn>
840 ifp1=0thenreturn:elseh=h+2:got
o830                                <fk>
850 p1=peek(p+b):p2=peek(c+b):poke
p-2+b,p1:pokec-2+b,p2               <dg>
860 ifp1=0thenh=h+0:goto830:elseb=
b+2:goto850                         <dc>
870 p1=peek(p+h):p2=peek(c+h):ifp1
=0then return                       <pn>
880 ifp1<>xthen920                  <pb>
890 ifp2<>ythenh=h+2:goto870        <ip>
900 p1=peek(p+2+h):p2=peek(c+2+h):
pokep+h,p1:pokec+h,p2:ifp1=0then r
eturn                               <km>
910 h=h+2:goto900                   <lc>
920 ifp1<>ythenh=h+2:goto870        <hi>
930 ifp2<>xthenh=h+2:goto870:else9
00                                  <ii>
940 p1=peek(p+h):p2=peek(c+h):ifp1
=0thenreturn                        <cl>
950 if(p1=x and p2=y) or (p2=x and
 p1=y)then pokep+h,l:pokec+h,b:ret
urn                                 <dl>
960 h=h+2:goto940                   <hd>
970 graphic0,1:input"name des obje
kts";a$:h=len(a$):fort=1toh         <bc>
980 poke1630+t,asc(mid$(a$,t,1)):n
ext:poke1630,h:return               <fm>
990 print"daten der punkte oder de
r linien":getkeyt$:ift$="l"then w=
0:goto1070                          <ep>
1000 w=1                            <en>
1010 graphic0,1:print"  daten alle
r punkte":n=peek(4096):t=0          <lh>
1020 print:print"  punkt    tiefe
  hoehe    breite"                  <jg>
1030 fort=0ton-1:y=2*t:a=q:gosub77
0:l=-x                              <fg>
1040 a=e:gosub770:h=-x:a=i:gosub77
0:b=x:print                         <mf>
1050 printt+1,l,h,b:poke205,peek(2
05)-1:poke201,peek(201)-1           <ii>
1060 getkeyt$:next:gosub1160:graph
ic0,1:ifw=0thenreturn               <hh>
1070 graphic0,1:print" daten aller
  linien":h=0                       <en>
1080 printchr$(17)"       vonCCCCCC
CCCCC>zu"                           <ha>
1090 p1=peek(p+h):p2=peek(c+h):h=h
+2:ifp1=0then1110                   <po>
1100 poke200,6:poke202,6:printp1;"
 ",p2:getkeyt$:goto1090             <af>
1110 ifw=1thenreturn:else1010       <al>
1120 p1=peek(p+h):p2=peek(c+h):ifp
1=0thenreturn                       <ei>
1130 ifp1>tthenpokep+h,p1-1         <lf>
1140 ifp2>tthenpokec+h,p2-1         <ij>
1150 h=h+2:goto1120                 <la>
1160 fort=0to30:next:return         <pm>
1170 graphic0,1:h=0:print" bewegun
gen def."                           <fp>
1180 print" ";h+1;".schritt   welc
```

```
he taste?(t.druecken)":gosub1160     <ao>
1190 lb=peek(198):iflb=64then1190:
else iflb=1thenpoke5465+h,255:goto
1230                                 <gg>
1200 ifpeek(198)=lbthen 1200:else
fort=0to9:poke1319+t,0:next          <fe>
1210 iflb=21orlb=17thenprint"  unz
ulaessig!!!":goto1180                <fm>
1220 input" fuer wiev. berechn.";h
b:poke5465+h,lb:poke5665+h,hb:h=h+
1:goto1180                           <lp>
1230 graphic0,1:print" daten der b
eweg.ablaeufe":h=0                   <co>
1240 poke208,peek(5465+h):ifpeek(2
08)=255then return                   <hj>
1250 getkeyt$:sys25650:lb=peek(208
)                                    <fp>
1260 iflb=128 thenprinth+1;"     cu
rsor rechts ",peek(5665+h):h=h+1:g
oto1240                              <nd>
1270 iflb=129 thenprinth+1;"     cu
rsor links  ",peek(5665+h):h=h+1:g
oto1240                              <pj>
1280 iflb=131 thenprinth+1;"     cu
rsor unten  ",peek(5665+h):h=h+1:g
oto1240                              <op>
1290 iflb=130 thenprinth+1;"     cu
rsor oben   ",peek(5665+h):h=h+1:g
oto1240                              <lc>
1300 printh+1;"     ";chr$(lb),peek
(5665+h):h=h+1:goto1240              <bj>
1310 graphic0,1:print" beweg.ablae
ufe aendern":h=0:gosub1230           <hk>
1320 print" der wievielte progr.sc
hritt soll":print"  geaendert werd
en"                                  <eb>
1330 inputh:print" neuer schritt":
fort=1319to1328:poket,0:next:a=pee
k(5464+h)                            <cg>
1340 gosub1160:lb=peek(198):iflb=6
4then1340                            <ii>
1350 gosub1160:ifpeek(198)=lbthen1
350:elsefort=1319to1328:poket,0:ne
xt                                   <ce>
1360 input" anzahl der berechnunge
n";hb:poke5464+h,lb:poke5664+h,hb    <gi>
1370 ifa=255thenpoke5465+h,255       <dh>
1380 print" noch mehr aendern j/n"
:getkeyt$:ift$="j"then 1310:else 5
70                                   <cd>
1390 q=4097:e=4353:i=4609:p=4865:c
=4866:return                         <mn>
==================================
P R O G R A M M E N D E
==================================
```

```
GRAFIX.CBM=======================p4
(p) COMMODORE WELT TEAM
===================================
(c) by Frank Koster

Plus4 (C16/116 + 64 KB)
===================================
```

Das Programm ist mit dem Maschinensprachmonitor TEDMON unter Zuhilfenahme des Pruefsummenprogrammes CHECKMON einzugeben und auf Diskette mit s"grafix.cbm",8,1000,1800 abzuspeichern, auf Kassette mit s"grafix.cbm",1,1000,1800.

```
===================================
>1000 30 24 80 24 80 00 00 28 :<c7>
>1008 00 70 00 70 00 28 00 00 :<81>
>1010 00 24 80 24 80 00 00 22 :<b0>
>1018 00 40 00 40 00 22 00 00 :<07>
>1020 00 24 80 24 80 00 00 28 :<d7>
>1028 00 70 00 70 00 28 00 00 :<c1>
>1030 00 24 80 24 80 00 00 22 :<70>
>1038 00 40 00 40 00 22 00 00 :<47>
>1040 00 24 80 84 80 5c 80 24 :<68>
>1048 80 24 80 84 80 5c 80 24 :<f8>
>1050 80 24 80 5c 80 84 80 24 :<24>
>1058 80 24 80 5c 80 84 80 24 :<14>
>1060 80 ff 00 ff 00 ff 00 ff :<a6>
>1068 00 ff 00 ff 00 ff 00 ff :<36>
>1070 00 ff 00 ff 00 ff 00 ff :<46>
>1078 00 ff 00 ff 00 ff 00 ff :<d5>
>1080 00 ff 00 ff 00 ff 00 ff :<66>
>1088 00 ff 00 ff 00 ff 00 ff :<76>
>1090 00 ff 00 ff 00 ff 00 ff :<86>
>1098 00 ff 00 ff 00 ff 00 ff :<96>
>10a0 00 ff 00 ff 00 ff 00 ff :<a6>
>10a8 00 ff 00 ff 00 ff 00 ff :<b6>
>10b0 00 ff 00 ff 00 ff 00 ff :<c6>
>10b8 00 ff 00 ff 00 ff 00 ff :<d6>
>10c0 00 ff 00 ff 00 ff 00 ff :<e6>
>10c8 00 ff 00 ff 00 ff 00 ff :<f6>
>10d0 00 ff 00 ff 00 ff 00 ff :<07>
>10d8 00 ff 00 ff 00 ff 00 ff :<17>
>10e0 00 ff 00 ff 00 ff 00 ff :<27>
>10e8 00 ff 00 ff 00 ff 00 ff :<37>
>10f0 00 ff 00 ff 00 ff 00 ff :<47>
>10f8 00 ff 00 ff 00 ff 00 01 :<57>
>1100 01 42 80 72 80 78 80 78 :<87>
>1108 80 32 80 32 00 78 00 78 :<c2>
>1110 00 72 00 42 00 48 00 48 :<4d>
>1118 00 2c 00 2c 80 48 80 48 :<ff>
>1120 80 42 80 72 80 78 80 78 :<c8>
>1128 80 32 80 32 00 78 00 78 :<82>
>1130 00 72 00 42 00 48 00 48 :<0d>
>1138 00 2c 00 2c 80 48 80 48 :<3e>
>1140 80 38 80 38 80 08 80 08 :<6a>
>1148 80 38 80 38 80 08 80 08 :<3a>
>1150 80 08 00 08 00 38 00 38 :<a5>
```

```
>1158 00 08 00 08 00 38 00 38 :<15>
>1160 00 ff 00 ff 00 ff 00 ff :<27>
>1168 00 ff 00 ff 00 ff 00 ff :<37>
>1170 00 ff 00 ff 00 ff 00 ff :<47>
>1178 00 ff 00 ff 00 ff 00 ff :<56>
>1180 00 ff 00 ff 00 ff 00 ff :<67>
>1188 00 ff 00 ff 00 ff 00 ff :<77>
>1190 00 ff 00 ff 00 ff 00 ff :<87>
>1198 00 ff 00 ff 00 ff 00 ff :<97>
>11a0 00 ff 00 ff 00 ff 00 ff :<a7>
>11a8 00 ff 00 ff 00 ff 00 ff :<b7>
>11b0 00 ff 00 ff 00 ff 00 ff :<c7>
>11b8 00 ff 00 ff 00 ff 00 ff :<d7>
>11c0 00 ff 00 ff 00 ff 00 ff :<e7>
>11c8 00 ff 00 ff 00 ff 00 ff :<f7>
>11d0 00 ff 00 ff 00 ff 00 ff :<08>
>11d8 00 ff 00 ff 00 ff 00 ff :<18>
>11e0 00 ff 00 ff 00 ff 00 ff :<28>
>11e8 00 ff 00 ff 00 ff 00 ff :<38>
>11f0 00 ff 00 ff 00 ff 00 ff :<48>
>11f8 00 ff 00 ff 00 ff 00 01 :<58>
>1200 00 18 00 18 00 18 00 18 :<f1>
>1208 00 18 00 18 00 18 00 18 :<20>
>1210 00 18 00 18 00 18 00 18 :<10>
>1218 00 18 00 18 00 18 00 18 :<42>
>1220 00 18 80 18 80 18 80 18 :<58>
>1228 80 18 80 18 80 18 80 18 :<c8>
>1230 80 18 80 18 80 18 80 18 :<b8>
>1238 80 18 80 18 80 18 80 18 :<28>
>1240 80 18 00 18 00 18 00 18 :<f2>
>1248 00 18 80 18 80 18 80 18 :<88>
>1250 80 18 00 18 00 18 00 18 :<12>
>1258 00 18 80 18 80 18 80 18 :<e8>
>1260 80 ff 00 ff 00 ff 00 ff :<a8>
>1268 00 ff 00 ff 00 ff 00 ff :<38>
>1270 00 ff 00 ff 00 ff 00 ff :<48>
>1278 00 ff 00 ff 00 ff 00 ff :<57>
>1280 00 ff 00 ff 00 ff 00 ff :<68>
>1288 00 ff 00 ff 00 ff 00 ff :<78>
>1290 00 ff 00 ff 00 ff 00 ff :<88>
>1298 00 ff 00 ff 00 ff 00 ff :<98>
>12a0 00 ff 00 ff 00 ff 00 ff :<a8>
>12a8 00 ff 00 ff 00 ff 00 ff :<b8>
>12b0 00 ff 00 ff 00 ff 00 ff :<c8>
>12b8 00 ff 00 ff 00 ff 00 ff :<d8>
>12c0 00 ff 00 ff 00 ff 00 ff :<e8>
>12c8 00 ff 00 ff 00 ff 00 ff :<f8>
>12d0 00 ff 00 ff 00 ff 00 ff :<09>
>12d8 00 ff 00 ff 00 ff 00 ff :<19>
>12e0 00 ff 00 ff 00 ff 00 ff :<29>
>12e8 00 ff 00 ff 00 ff 00 ff :<39>
>12f0 00 ff 00 ff 00 ff 00 ff :<49>
>12f8 00 ff 00 ff 00 ff 00 01 :<59>
>1300 01 01 02 01 10 01 11 02 :<8c>
>1308 03 02 12 03 04 03 13 04 :<f1>
>1310 05 04 14 05 06 05 15 06 :<01>
>1318 07 06 16 07 08 07 17 08 :<12>
>1320 09 08 18 09 0a 09 19 0a :<12>
>1328 0b 0a 1a 0b 0c 0b 1b 0c :<32>
>1330 0d 0c 1c 0d 0e 0d 1d 0e :<01>
>1338 0f 0e 1e 0f 10 0f 1f 10 :<52>
>1340 20 12 13 13 14 14 15 15 :<4c>
>1348 16 16 17 17 18 18 19 19 :<4e>
>1350 1a 1a 1b 1b 1c 1c 1d 1d :<5e>
>1358 1e 1e 1f 1f 20 11 12 11 :<c4>
>1360 20 21 22 21 24 21 25 22 :<88>
>1368 23 22 26 23 24 23 27 24 :<5d>
>1370 28 25 26 25 28 26 27 27 :<4a>
>1378 28 29 2a 29 2c 29 2d 2a :<b7>
>1380 2b 2a 2e 2b 2c 2b 2f 2c :<ab>
>1388 30 2d 2e 2e 2f 2f 30 2d :<0b>
>1390 30 00 00 ff 00 ff 00 ff :<bb>
>1398 00 ff 00 ff 00 ff 00 ff :<99>
>13a0 00 ff 00 ff 00 ff 00 ff :<a9>
>13a8 00 ff 00 ff 00 ff 00 ff :<b9>
>13b0 00 ff 00 ff 00 ff 00 ff :<c9>
>13b8 00 ff 00 ff 00 ff 00 ff :<d9>
>13c0 00 ff 00 ff 00 ff 00 ff :<e9>
>13c8 00 ff 00 ff 00 ff 00 ff :<f9>
>13d0 00 ff 00 ff 00 ff 00 ff :<0a>
>13d8 00 ff 00 ff 00 ff 00 ff :<1a>
>13e0 00 ff 00 ff 00 ff 00 ff :<2a>
>13e8 00 ff 00 ff 00 ff 00 ff :<3a>
>13f0 00 ff 00 ff 00 ff 00 ff :<4a>
>13f8 00 ff 00 ff 00 ff 00 01 :<5a>
>1400 01 e7 00 ff 00 ff 00 ff :<3a>
>1408 00 ff 00 ff 00 ff 00 ff :<79>
>1410 00 ff 00 ff 00 ff 00 ff :<89>
>1418 00 ff 00 ff 00 ff 00 ff :<99>
>1420 00 ff 00 ff 00 ff 00 ff :<a9>
>1428 00 ff 00 ff 00 ff 00 ff :<b9>
>1430 00 ff 00 ff 00 ff 00 ff :<c9>
>1438 00 ff 00 ff 00 ff 00 ff :<d9>
>1440 00 ff 00 ff 00 ff 00 ff :<e9>
>1448 00 ff 00 ff 00 ff 00 ff :<f9>
>1450 00 ff 00 ff 00 ff 00 ff :<0a>
>1458 00 ff 00 ff 00 ff 00 ff :<1a>
>1460 00 ff 00 ff 00 ff 00 ff :<2a>
>1468 00 ff 00 ff 00 ff 00 ff :<3a>
>1470 00 ff 00 ff 00 ff 00 ff :<4a>
>1478 00 ff 00 ff 00 ff 00 ff :<59>
>1480 00 ff 00 ff 00 ff 00 ff :<6a>
>1488 00 ff 00 ff 00 ff 00 ff :<7a>
>1490 00 ff 00 ff 00 ff 00 ff :<8a>
>1498 00 ff 00 ff 00 ff 00 ff :<9a>
>14a0 00 ff 00 ff 00 ff 00 ff :<aa>
>14a8 00 ff 00 ff 00 ff 00 ff :<ba>
>14b0 00 ff 00 ff 00 ff 00 ff :<ca>
>14b8 00 ff 00 ff 00 ff 00 ff :<da>
>14c0 00 ff 00 ff 00 ff 00 ff :<ea>
>14c8 00 ff 00 ff 00 ff 00 ff :<fa>
>14d0 00 ff 00 ff 00 ff 00 ff :<0b>
>14d8 00 ff 00 ff 00 ff 00 ff :<1b>
>14e0 00 ff 00 ff 00 ff 00 ff :<2b>
>14e8 00 ff 00 ff 00 ff 00 ff :<3b>
>14f0 00 ff 00 ff 00 ff 00 ff :<4b>
```

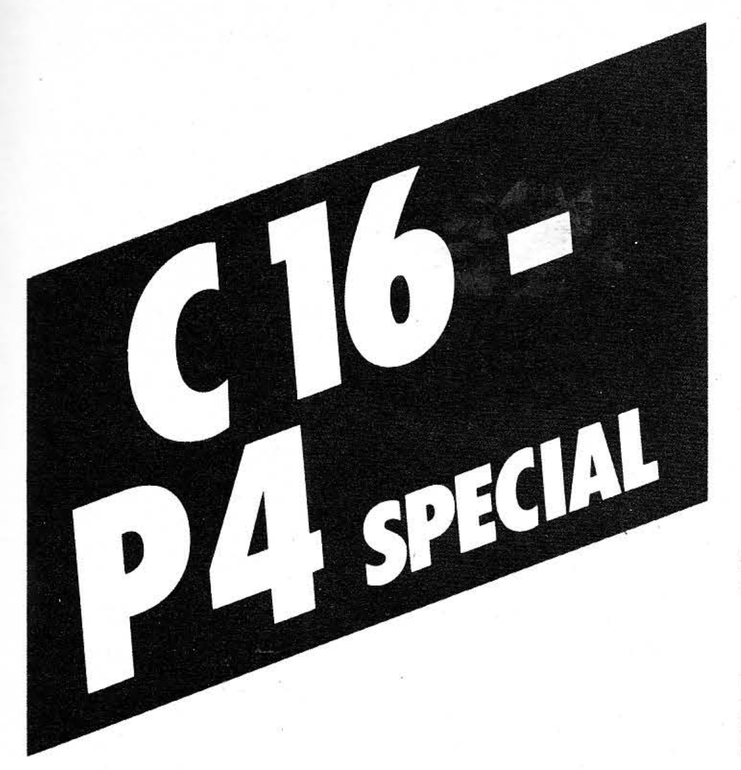
```
>14f8 00 ff 00 ff 00 ff 00 01 :<5b>
>1500 01 e7 00 ff 00 ff 00 ff :<3b>
>1508 00 ff 00 ff 00 ff 00 ff :<7a>
>1510 00 ff 00 ff 00 ff 00 ff :<8a>
>1518 00 ff 00 ff 00 ff 00 ff :<9a>
>1520 00 ff 00 ff 00 ff 00 ff :<aa>
>1528 00 ff 00 ff 00 ff 00 ff :<ba>
>1530 00 ff 00 ff 00 ff 00 ff :<ca>
>1538 00 ff 00 ff 00 ff 00 ff :<da>
>1540 00 ff 00 ff 00 ff 00 ff :<ea>
>1548 00 ff 00 ff 00 ff 00 ff :<fa>
>1550 00 ff 00 ff 00 ff 00 ff :<0b>
>1558 00 23 16 28 33 ff 00 ff :<36>
>1560 00 ff 00 ff 00 ff 00 ff :<2b>
>1568 00 ff 00 ff 00 ff 00 ff :<3b>
>1570 00 ff 00 ff 00 ff 00 ff :<4b>
>1578 00 ff 00 ff 00 ff 00 ff :<5a>
>1580 00 ff 00 ff 00 ff 00 ff :<6b>
>1588 00 ff 00 ff 00 ff 00 ff :<7b>
>1590 00 ff 00 ff 00 ff 00 ff :<8b>
>1598 00 ff 00 ff 00 ff 00 ff :<9b>
>15a0 00 ff 00 ff 00 ff 00 ff :<ab>
>15a8 00 ff 00 ff 00 ff 00 ff :<bb>
>15b0 00 ff 00 ff 00 ff 00 ff :<cb>
>15b8 00 ff 00 ff 00 ff 00 ff :<db>
>15c0 00 ff 00 ff 00 ff 00 ff :<eb>
>15c8 00 ff 00 ff 00 ff 00 ff :<fb>
>15d0 00 ff 00 ff 00 ff 00 ff :<0c>
>15d8 00 ff 00 ff 00 ff 00 ff :<1c>
>15e0 00 ff 00 ff 00 ff 00 ff :<2c>
>15e8 00 ff 00 ff 00 ff 00 ff :<3c>
>15f0 00 ff 00 ff 00 ff 00 ff :<4c>
>15f8 00 ff 00 ff 00 ff 00 01 :<5c>
>1600 01 e7 00 ff 00 ff 00 ff :<3c>
>1608 00 ff 00 ff 00 ff 00 ff :<7b>
>1610 00 ff 00 ff 00 ff 00 ff :<8b>
>1618 00 ff 00 ff 00 ff 00 ff :<9b>
>1620 00 01 01 08 c8 ff 00 ff :<78>
>1628 00 ff 00 ff 00 ff 00 ff :<bb>
>1630 00 ff 00 ff 00 ff 00 ff :<cb>
>1638 00 ff 00 ff 00 ff 00 ff :<db>
>1640 00 ff 00 ff 00 ff 00 ff :<eb>
>1648 00 ff 00 ff 00 ff 00 ff :<fb>
>1650 00 ff 00 ff 00 ff 00 ff :<0c>
>1658 00 ff 00 ff 00 ff 00 ff :<1c>
>1660 00 ff 00 ff 00 ff 00 ff :<2c>
>1668 00 ff 00 ff 00 ff 00 ff :<3c>
>1670 00 ff 00 ff 00 ff 00 ff :<4c>
>1678 00 ff 00 ff 00 ff 00 ff :<5b>
>1680 00 ff 00 ff 00 ff 00 ff :<6c>
>1688 00 ff 00 ff 00 ff 00 ff :<7c>
>1690 00 ff 00 ff 00 ff 00 ff :<8c>
>1698 00 ff 00 ff 00 ff 00 ff :<9c>
>16a0 00 ff 00 ff 00 ff 00 ff :<ac>
>16a8 00 ff 00 ff 00 ff 00 ff :<bc>
>16b0 00 ff 00 ff 00 ff 00 ff :<cc>
>16b8 00 ff 00 ff 00 ff 00 ff :<dc>
>16c0 00 ff 00 ff 00 ff 00 ff :<ec>
>16c8 00 ff 00 ff 00 ff 00 ff :<fc>
>16d0 00 ff 00 ff 00 ff 00 ff :<0d>
>16d8 00 ff 00 ff 00 ff 00 ff :<1d>
>16e0 00 ff 00 ff 00 ff 00 ff :<2d>
>16e8 00 ff 00 ff 00 ff 00 ff :<3d>
>16f0 00 ff 00 ff 00 ff 00 ff :<4d>
>16f8 00 ff 00 ff 00 ff 00 01 :<5d>
>1700 01 e7 00 ff 00 ff 00 ff :<3d>
>1708 00 ff 00 ff 00 ff 00 ff :<7c>
>1710 00 ff 00 ff 00 ff 00 ff :<8c>
>1718 00 ff 00 ff 00 ff 00 ff :<9c>
>1720 00 ff 00 ff 00 ff 00 ff :<ac>
>1728 00 ff 00 ff 00 ff 00 ff :<bc>
>1730 00 ff 00 ff 00 ff 00 ff :<cc>
>1738 00 ff 00 ff 00 ff 00 ff :<dc>
>1740 00 ff 00 ff 00 ff 00 ff :<ec>
>1748 00 ff 00 ff 00 ff 00 ff :<fc>
>1750 00 ff 00 ff 00 ff 00 ff :<0d>
>1758 00 ff 00 ff 00 ff 00 ff :<1d>
>1760 00 ff 00 ff 00 ff 00 ff :<2d>
>1768 00 ff 00 ff 00 ff 00 ff :<3d>
>1770 00 ff 00 ff 00 ff 00 ff :<4d>
>1778 00 ff 00 ff 00 ff 00 ff :<5c>
>1780 00 ff 00 ff 00 ff 00 ff :<6d>
>1788 00 ff 00 ff 00 ff 00 ff :<7d>
>1790 00 ff 00 ff 00 ff 00 ff :<8d>
>1798 00 ff 00 ff 00 ff 00 ff :<9d>
>17a0 00 ff 00 ff 00 ff 00 ff :<ad>
>17a8 00 ff 00 ff 00 ff 00 ff :<bd>
>17b0 00 ff 00 ff 00 ff 00 ff :<cd>
>17b8 00 ff 00 ff 00 ff 00 ff :<dd>
>17c0 00 ff 00 ff 00 ff 00 ff :<ed>
>17c8 00 ff 00 ff 00 ff 00 ff :<ff>
>17d0 00 ff 00 ff 00 ff 00 ff :<0e>
>17d8 00 ff 00 ff 00 ff 00 ff :<1e>
>17e0 00 ff 00 ff 00 ff 00 ff :<2e>
>17e8 00 ff 00 ff 00 ff 00 ff :<3e>
>17f0 00 ff 00 ff 00 ff 00 ff :<4e>
>17f8 00 ff 00 ff 00 ff 00 01 :<5e>
==================================
P R O G R A M M E N D E
==================================
```

Fortsetzung von Seite 51

GESCHWINDIGKEITS-STEIGERUNG

Die Rechengeschwindigkeit im L-Modus kann erhöht werden, wenn der zu löschende Bereich des Grafikbildschirm verkleinert wird. Dies kann geschehen durch:
POKE DEC("6D45"),HA
POKE DEC("6D26"),HE

Mit HA sei hierbei das Highbyte der Adresse, ab der gelöscht werden soll, bezeichnet. HE sei das Highbyte der Adresse, bis zu der gelöscht werden soll. Es ist zu beachten, daß HA größer oder gleich 20 zu sein hat und HE kleiner gleich 63. □



KOMMT REGELMÄSSIG ZU IHNEN INS HAUS

Finden Sie Ihr C16/P4 nicht am Kiosk? Weil es schon ausverkauft ist? Oder „Ihr" Kiosk nicht beliefert wurde? Kein Problem! Für ganze 60 DM liefern wir Ihnen per Post sechs Hefte ins Haus (Ausland 80 DM). Einfach den Bestellschein ausschneiden – fotokopieren oder abschreiben, in einen Briefumschlag und ab per Post (Achtung: Porto nicht vergessen). C16/P4-SPECIAL kommt dann pünktlich ins Haus.

WICHTIGE RECHTLICHE GARANTIE!
Sie können diesen Abo-Auftrag binnen einer Woche nach Eingang der Abo-Bestätigung durch den Verlag widerrufen – Postkarte genügt. Zur Wahrung der Frist genügt die rechtzeitige Absendung. Ansonsten läuft dieser Auftrag jeweils für sechs Ausgaben, wenn ihm nicht vier Wochen vor Ablauf widersprochen wird, weiter.

DAS SONDERANGEBOT: PRIVATE KLEINANZEIGEN KOSTENLOS!

Das bietet Ihnen COMMODORE-WELT: KLEINANZEIGEN SIND KOSTENLOSE FÜR PRIVATANBIETER! Suchen Sie etwas, haben Sie etwas zu verkaufen, zu tauschen, wollen Sie einen Club gründen? Coupon ausfüllen, auf Postkarte kleben oder in Briefumschlag stecken und abschicken. So einfach geht das. Wollen Sie das Heft nicht zerschneiden, können Sie den Coupon auch fotokopieren. Oder einfach den Anzeigentext uns so schicken, auf Postkarte oder im Brief. Aber bitte mit Druckbuchstaben oder in Schreibmaschinenschrift!
Und: Einschließlich Ihrer Adresse und/oder Telefonnummer sollten acht Zeilen à 28 Anschläge nicht überschritten werden.
ACHTUNG: WICHTIGER HINWEIS!
Wir veröffentlichen nur Kleinanzeigen privater Inserenten, keine gewerblichen Anzeigen. Die kosten pro Millimeter DM 5.00 plus Mehrwertsteuer!
Wir versenden für Privat-Inserenten keine Beleg-Exemplare!
Chiffre-Anzeigen sind nicht gestattet! Wir behalten uns vor, Anzeigen, die gegen rechtliche, sittliche oder sonstige Gebote verstoßen, abzulehnen!
Anzeigenabdruck in der Reihenfolge ihres Eingangs, kein Rechtsanspruch auf den Abdruck in der nächsten Ausgabe!
Die Insertion ist nicht vom Kauf des Heftes abhängig! Wir behalten uns vor, Anzeigen, die nicht zum Themenkreis des Heftes – Computer – gehören, nicht abzudrucken oder sie nur insoweit zu berücksichtigen, wie es der Umfang des kostenlosen Anzeigenteils zuläßt.

ANZEIGENSERVICE

Die große Börse für jeden Zweck in der CBM REVUE / COMMODORE-WELT. Kostenlos für Privat-Inserenten. Spottbillig für gewerbliche Anbieter. Einfach Coupon ausschneiden, fotokopieren o.ä., ausfüllen und ab die Post – Freimachen nicht vergessen! – Unsere Adresse steht auf dem Coupon, ebenso die Preise für gewerbliche Anbieter! Achtung! Wir weisen ausdrücklich darauf hin, daß wir offensichtlich gewerbliche Anzeigen nicht kostenlos veröffentlichen und uns jedweden Abdruck kostenloser Anzeigen vorbehalten müssen, insbesondere, wenn deren Inhalt nicht CBM-typisch ist oder gegen geltendes Recht verstößt. Private Chiffreanzeigen werden nicht aufgenommen. Für Privatanbieter: maximal acht Zeilen à 28 Anschläge. Für gewerbliche Anbieter: 5 DM p. mm.

COMMODORE WELT
ANZEIGENABTLG
POSTFACH 1161
D-8044 UNTERSCHLEISSHEIM

Name ____________________

Vorname ____________________

Straße/Hausnr. ____________________

Plz/Ort ____________________

ABO-SERVICE-KARTE

16/I **COUPON**

Ja, ich möchte von Ihrem Angebot Gebrauch machen.
Bitte senden Sie mir bis auf Widerruf ab sofort jeweils die nächsten sechs Ausgaben an untenstehende Anschrift. Wenn ich nicht vier Wochen vor Ablauf kündige, läuft diese Abmachung automatisch weiter.

Ich nehme zur Kenntnis, daß die Belieferung erst beginnt, wenn die Abo-Gebühr dem Verlag zugegangen ist.

Name ____________________ **16/I**

Vorname ____________________

Straße/Hausnr. ____________________

Plz/Ort ____________________

Ich bezahle:

- ☐ per beiliegendem Verrechnungsscheck
- ☐ gegen Rechnung
- ☐ bargeldlos per Bankeinzug von meinem Konto

bei (Bank) und Ort ____________________

Kontonummer ____________________

Bankleitzahl ____________________
(steht auf jedem Kontoauszug)

Unterschrift ____________________

Von meinem Widerspruchsrecht habe ich Kenntnis genommen.

Unterschrift ____________________

COMMODORE WELT
ABO-SERVICE **16/I**
POSTFACH 1161
D-8044
UNTERSCHLEISSHEIM

PROGRAMMSERVICE

Hiermit bestelle ich in Kenntnis Ihrer Verkaufsbedingungen die Listings dieses Heftes auf

☐ **Kassetten zu 40,-** ☐ **Disketten zu 40,- (16er)**

Ich zahle:
Zutreffendes bitte ankreuzen!
per beigefügtem Scheck ( ) Schein ( )

Gegen Bankabbuchung am Versandtag ( )

Meine Bank (mit Ortsname) ______________________ **16/I**

Meine Kontonummer ______________________

Meine Bankleitzahl ______________________ (steht auf jedem Bankauszug) ______________________

Vorname ______________________ Nachname ______________________

Str./Nr. ______________________ Plz/Ort ______________________

Verkaufsbedingungen: Lieferung nur gegen Vorkasse oder Bankabbuchung. Keine Nachnahme. Umtausch bei Nichtfunktionieren. **16/I**

Unterschrift ______________________

Bitte ausschneiden und einsenden an

COMMODORE WELT
KASSETTENSERVICE **16/I**
POSTFACH 1161
D-8044 UNTERSCHLEISSHEIM

LESER WERBEN LESER

GEWINNEN SIE EINE COMPUTER-UHR! Und zusätzlich eventuell noch ein großes Commodore-Buch. Oder ein Paket Disketten. ODER AUCH EINEN COMMODORE-DRUCKER – ODER EINE DISKETTENSTATION! Wie? Sie werben einen Abonnenten. Dann haben Sie auf jeden Fall schon die Computer-Uhr gewonnen. Zusätzlich verlosen wir unter allen, die mitmachen, jeden Monat vier weitere wertvolle Preise. Und alle sechs Monate gibt es einen Hauptpreis unter allen Abo-Werbern zu gewinnen. Also: Mitmachen. Mitgewinnen.

Ja, ich mache mit beim Abo-Wettbewerb. Ich habe



Herrn/Frau ______________________

____

Straße/Hausnr. ______________________

____

Plz/Ort ______________________

____

als neuen Abonnenten der COMMODORE WELT geworben.

Der neue Abonnent war bisher noch nicht Bezieher dieser Zeitschrift.
Als Prämie erhalte ich nach Eingang des Abo-Entgeltes auf jeden Fall eine Computer-Uhr, wie abgebildet, und nehme zusätzlich noch an der Verlosung des Monats sowie der halbjährlichen Hauptpreise teil. Mir ist bekannt, daß der Rechtsweg bei den Verlosungen ausgeschlossen ist.
Meinen Preis senden Sie an

Name ______________________

Straße/Hsnr. ______________________

Plz/Ort ______________________

(Bitte ausschneiden und zusammen mit der Abo-Bestellkarte links einsenden!) **16/I**

VERDIENEN SIE GELD MIT IHREM COMPUTER!

Haben Sie einen Commodore VC 20 oder C 64? Einen 16/116, Plus 4? Oder einen 128? Können Sie programmieren? In Basic oder Maschinensprache? Dann bietet COMMODORE-WELT Ihnen die Möglichkeit, mit diesem Hobby Geld zu verdienen!

Wie? Ganz einfach. Sie senden uns die Programme, die Sie für einen Abdruck als geeignet halten, zusammen mit einer Kurzbeschreibung, aus der auch die verwendete Hardware – eventuelle Erweiterungen – benutzte Peripherie – hervorgehen muß (Schauen Sie sich dazu den Kopf unserer Programmlistings an.)

Benötigt werden: Zwei Listings des Programms sowie eine Datenkassette oder Diskette! Wenn die Redaktion sich überzeugt hat, daß dieses Programm läuft und sich zum Abdruck eignet, zahlen wir Ihnen pro Programm je nach Umfang bis zu DM 300,–!

Sollten Sie keinen Drucker haben, genügt der Datenträger.

Sie erhalten Ihre Kassette/Diskette selbstverständlich zurück, wenn Sie einen ausreichend frankierten Rückumschlag mit Ihrer Adresse beifügen.

Bei der Einsendung müssen Sie mit Ihrer Unterschrift garantieren, daß Sie der alleinige Inhaber der Urheber-Rechte sind! Benutzen Sie bitte anhängendes Formular! (Wir weisen darauf hin, daß auch die Redaktion amerikanische und englische Fachzeitschriften liest und „umgestaltete" Programme ziemlich schnell erkennt).

Um Ihnen die Arbeit zu erleichtern, finden Sie hier ein Formular. Sie können es ausschneiden oder fotokopieren.

Name des Einsenders: ______________________________
Straße/Hausnr./Tel.: ______________________________
Plz/Ort: ______________________________

Hiermit biete ich Ihnen zum Abdruck folgende(s) Programm(e) an:
____
____
____
____

Benötigte Geräte: ______________________________
____

Beigefügt ( ) Listings ( ) Kassette ( ) Diskette

Ich versichere, der alleinige Urheber des Programmes zu sein!
Hiermit ermächtige ich die Redaktion, dieses Programm abzudrucken und wirtschaftlich zu verwerten. Sollte es in den Kassetten-Service aufgenommen werden, erhalte ich auch dafür eine entsprechende Vergütung, das Copyright geht insoweit auf den Verlag über.

____
Rechtsverbindliche Unterschrift

COMMODORE WELT
PROGRAMM-REDAKTION
POSTFACH 1161
D-8044 UNTERSCHLEISSHEIM

Botanik

In einer Zeit, in der MacDonald's-Restaurants und Imbißbuden wie Pilze aus dem Boden schießen, darf eines nicht vernachlässigt werden, nämlich das Wissen um den Gartenbau und um gesunde Ernährung. Testen Sie Ihre Kenntnisse.

Je zehn Fragen zu den fünf Themen
– Allgemeiner Gartenbau
– Gemüsebau
– Obstbau
– Verwendung, Haushalt
– Biologischer Anbau
warten auf Ihre Beantwortung. Drei Antworten sind jeweils zur Wahl gestellt. Nachdem Sie den Test absolviert haben, bewertet das Programm Ihre Leistungen. Auf Tastendruck können Sie sowohl die richtigen als auch Ihre unrichtigen Antworten, schwarz auf weiß ausgedruckt, erfahren. □

```
10 rem botaniktest =============p4 <jl>
20 rem (p) commodore welt team <ho>
30 rem ========================== <ng>
40 rem (c) by hermann wellesen 7.0 <bf>
50 rem (v) by bernd welte      3.5 <mg>
60 rem <ah>
70 rem basic v3.5 <nl>
80 rem plus4 (c16/116 + 64 kb) <fd>
90 rem ========================== <jg>
100 rem ************************** <hl>
110 rem * angeregt durch die      * <pc>
120 rem * landwirtschaftskammer   * <de>
130 rem * rheinland               * <la>
140 rem ************************** <ok>
150 s1$=chr$(032):bs$=chr$(045) <hh>
160 a2$=s1$+s1$:a3$=a2$+s1$ <dh>
170 a4$=a3$+s1$:a5$=a4$+s1$ <ik>
180 cl$=chr$(147):c4$=chr$(017) <jf>
190 c1$=chr$(157):rn$=chr$(018) <gj>
200 rf$=chr$(146) <mo>
210 poke65305,0:poke65301,0 <go>
220 rem * run/stop sperren, aber * <bd>
230 rem * nicht run/stop/restore * <jm>
240 printchr$(014);chr$(008);chr$(
155) <ik>
250 rem ********* dimensionieren * <im>
260 rem * und einlesen der fragen* <bg>
270 dima$(250):fori=1to250 <lj>
280 reada$(i):nexti <hl>
290 rem * definition sachgebiete * <em>
300 t$(1)="Allgemeiner Gartenbau" <jm>
310 t$(2)="Gemuesebau" <kk>
320 t$(3)="Obstbau" <gk>
330 t$(4)="Verwendung, Haushalt" <ll>
340 t$(5)="Biologischer Anbau" <lb>
350 rem ************* definition * <nh>
360 rem *** der bewertungssaetze * <ma>
370 en$(0)="Ihr Wissen laesst zu w
uenschen uebrig" <pj>
380 en$(1)=en$(0) <on>
390 en$(2)=en$(0) <ka>
400 en$(3)=en$(0) <mj>
410 en$(4)="Na ja, das koennte bes
ser sein" <hp>
420 en$(5)=en$(4) <gn>
430 en$(6)="Schon ganz gut, was Si
e so wissen" <bl>
440 en$(7)=en$(6) <ok>
450 en$(8)="Das ist ja prima. Fast
 volle Punktzahl" <lc>
460 en$(9)=en$(8) <gm>
470 en$(10)="Bravo, alle Antworten
 richtig." <jj>
480 rem *** bewertungsvariabelen * <oa>
490 rem ********* zuruecksetzten * <oc>
500 fori=0to10:b(i)=0:nexti:b=0 <ac>
510 rem ************* hauptmenue * <km>
520 printcl$c4$c4$c4$a2$"Was wisse
n Sie vom Gartenbau ?" <om>
530 printc4$c4$a2$"Hier koennen Si
e an 10 Fragen Ihr" <oh>
540 printa2$"Allgemeinwissen ueber
pruefen."c4$ <lg>
550 fori=1to5 <jn>
560 printc4$a2$rn$;i;c1$s1$rf$s1$t
$(i) <bd>
570 nexti <do>
580 printc4$c4$a3$rn$" Waehlen Sie
 Ihr Fachgebiet ! "rf$ <kc>
590 getz$:z=val(z$) <op>
600 ifz<1orz>5then590 <je>
610 rem ******* beginn des tests * <eb>
620 v=(z-1)*50 <jl>
630 fori=0to45step5:ii=i+v <el>
640 printcl$rn$" Es ist immer nur
eine Antwort richtig! "rf$ <id>
650 rem *********** fragestellung * <pg>
660 printa$(ii+1):printa$(ii+2) <pm>
670 rem ************* ausgabe der * <lk>
680 rem *** moeglichen antworten * <an>
690 forj=3to5 <mo>
700 printc4$s1$rn$;j-2;c1$s1$rf$c4
$ <ac>
710 printleft$(a$(ii+j),len(a$(ii+
j))-1);c4$ <dn>
720 ifright$(a$(ii+j),1)<>s1$thenc
=val(right$(a$(ii+j),1)) <lm>
730 nextj <ma>
740 rem ********* antwortabfrage * <oj>
750 getz$:z=val(z$) <pe>
760 ifz<1orz>3then750 <oh>
770 ifz=cthenb=b+1:gosub1140 <oj>
780 ifz<>cthenb(i/5)=ii+2+z:gosub1
180 <dn>
790 nexti <lc>
800 rem ******** ergebnisausgabe * <mo>
810 rem *********** mit bewertung * <md>
820 printcl$c4$c4$c4$c4$a5$a3$rn$"
 Und hier Ihr Ergebnis: "rf$ <jk>
830 printc4$"Von 10 Fragen zum The
ma" <nh>
840 printc4$rn$s1$t$(v/50+1)s1$rf$ <en>
850 printc4$"wurden";b;"Fragen ric
htig beantwortet." <cp>
860 printc4$c4$c4$"Zu dieser Leist
ung sage ich:" <pg>
870 printc4$en$(b) <bi>
880 printc4$c4$c4$c4$a4$rn$" 1 "rf
$" = Neustart"a5$; <db>
890 printrn$" 0 "rf$;" = Drucken" <cj>
900 getz$ <jl>
910 ifz$<>"1"andz$<>"0"then900 <fa>
920 ifz$="1"then500 <oh>
930 rem ***** drucken der fragen * <jl>
940 rem * antworten & bewertungen* <mn>
950 open1,4:cmd1:print <jg>
```

```
960 printa5$s1$"Sie testeten Ihr W
issen zum Thema "t$(v/50+1)          <gf>
970 print:fori=0to45step5:ii=i+v:p
rint                                 <fl>
980 ifright$(a$(ii+1),1)<>bs$thenp
rinta$(ii+1)s1$a$(ii+2):goto1000     <ij>
990 printleft$(a$(ii+1),len(a$(ii+
1))-1);a$(ii+2)                      <gm>
1000 forj=3to5                       <ii>
1010 ifright$(a$(ii+j),1)<>s1$then
printleft$(a$(ii+j),len(a$(ii+j))-
1)                                   <jp>
1020 nextj                           <en>
1030 ifb(i/5)<>0thenprint"Ihre Ant
wort lautete "a$(b(i/5))             <aj>
1040 print:nexti:print:print         <if>
1050 printchr$(14)"Und hier Ihr Er
gebnis:"chr$(15)                     <ad>
1060 printc4$"Von 10 Fragen zum Th
ema "t$(v/50+1)                      <ij>
1070 print"wurden";b;"Fragen richt
ig beantwortet."                     <gd>
1080 print                           <bb>
1090 print"Zu dieser Leistung sage
 ich:  "en$(b)                       <ch>
1100 print#1                         <kb>
1110 close1:goto500                  <ig>
1120 rem ********* unterprogramm *   <ma>
1130 rem ****** richtige antwort *   <ei>
1140 printc4$a4$rn$" Bravo, die An
twort ist richtig "rf$               <bk>
1150 fork=1to1000:nextk:return       <fj>
1160 rem ********* unterprogramm *   <jo>
1170 rem ******* falsche antwort *   <bo>
1180 printa2$rn$" Die Antwort ist
leider falsch "rf$                   <mc>
1190 printc4$" Richtig waere : "rn
$;c;c1$s1$rf$                        <ce>
1200 fork=1to2000:nextk:return       <nf>
1210 rem * datastatements mit den*   <cj>
1220 rem ** fragen und antworten *   <ap>
1230 rem * allgemeiner gartenbau *   <ap>
1240 data"Wieviel gaertnerische Be
triebe gibt"                         <en>
1250 data"es in der Bundesrepublic
k Deutschland ?"                     <kj>
1260 data"15000 "                    <ni>
1270 data"30000 "                    <ko>
1280 data"450003"                    <bp>
1290 data"Mit wieviel % ist der Ga
rtenbau am"                          <mm>
1300 data"Gesamtenergieverbrauch b
eteiligt ?"                          <kf>
1310 data"0.5 %1"                    <be>
1320 data"1.0 % "                    <gf>
1330 data"2.3 % "                    <go>
1340 data"Wie gross ist die Gemues
eanbauflaeche"                       <lk>
1350 data"in Nordrhein-Westfalen ?
"                                    <ei>
1360 data" 2500 ha "                 <hl>
1370 data" 8000 ha2"                 <jl>
1380 data"10000 ha "                 <ln>
1390 data"Welches Bundesland hat d
ie groesste"                         <in>
1400 data"Anbauflaeche fuer Gemues
e ?"                                 <cb>
1410 data"Nordrhein-Westfalen1"      <kh>
1420 data"Bayern "                   <ap>
1430 data"Niedersachsen "            <bk>
1440 data"Wieviel Stueck Kopfsalat
 werden in der"                      <in>
1450 data"Bundesrepublik Deutschla
nd produziert ?"                     <ha>
1460 data"100 Mio. Stueck "          <bi>
1470 data"140 Mio. Stueck "          <bc>
1480 data"180 Mio. Stueck3"          <no>
1490 data"Wodurch kann die heimisc
he Gemuese-"                         <jo>
1500 data"saison verlaengert werde
n ?"                                 <mo>
1510 data"Anbau im Gewaechshaus1"    <bn>
1520 data"verstaerkte Importe "      <ic>
1530 data"mehr Arbeitskraefte "      <pm>
1540 data"Wieviel Obstbaubetriebe
gibt es"                             <ek>
1550 data"im Rheinland ?"            <cc>
1560 data" 7301"                     <hm>
1570 data"1150 "                     <cn>
1580 data"4750 "                     <ia>
1590 data"Welche der angebauten Ob
starten nimmt"                       <bp>
1600 data"die groesste Anbauflaech
e ein ?"                             <pc>
1610 data"Erdbeeren "                <ao>
1620 data"Aepfel2"                   <lo>
1630 data"Sauerkirschen "            <om>
1640 data"Wie gross ist die gesamt
e Gewaechs-"                         <dj>
1650 data"hausflaeche in der Bunde
srepublik ?"                         <nm>
1660 data"1500 ha "                  <mo>
1670 data"2700 ha "                  <el>
1680 data"3400 ha3"                  <ac>
1690 data"Wieviel Arbeitskraefte b
eschaeftigt"                         <ni>
1700 data"der Deutsche Gartenbau ?
"                                    <me>
1710 data" 95000 "                   <bn>
1720 data"1750002"                   <kg>
1730 data"240000 "                   <pp>
1740 rem ************ gemuesebau *   <de>
1750 data"Wie hoch ist der jaehrli
che Pro- Kopf- "                     <cd>
1760 data"Verbrauch an Gemuese in
Deutschland ?"                       <np>
```

```
1770 data" 70 Kg1"                         <hk>
1780 data" 90 Kg "                         <dd>
1790 data"110 Kg "                         <nm>
1800 data"In welchem EG-Land wird
die"                                       <lh>
1810 data"dreifache Menge verzehrt
 ?"                                        <pl>
1820 data"Griechenland1"                   <db>
1830 data"Frankreich "                     <pn>
1840 data"Italien "                        <oj>
1850 data"Ab wann ist das erste he
imische"                                   <lc>
1860 data"Freilandgemuese auf dem
Markt ?"                                   <kd>
1870 data"Ende April/Anfang Mai1"          <eo>
1880 data"Ende Mai/Anfang Juni "           <bg>
1890 data"Ende Juni/Anfang Juli "          <jd>
1900 data"Wann ist die Saison fuer
 den"                                      <gd>
1910 data"heimischen Spargel ?"            <jl>
1920 data"Februar - Maerz "                <ib>
1930 data"April - Juni2"                   <ej>
1940 data"Juli - September "               <fg>
1950 data"Welches Land, ist das ur
spruengliche"                              <ig>
1960 data"Heimatland der Tomate ?"         <ik>
1970 data"Spanien "                        <ph>
1980 data"Suedasien "                      <la>
1990 data"Suedamerika3"                    <ji>
2000 data"Was ist Pak Choy "               <df>
2010 data"(oder Paksoi) ?"                 <gh>
2020 data"eine Bonsaiart "                 <om>
2030 data"ein Chinesischer Gemuese
baubetrieb "                               <ij>
2040 data"neue, aus Japan stammend
e Kohlart, die  auch im Rheinland
angebaut wird3"                            <ik>
2050 data"Werden Brokkoli und Fenc
hel auch"                                  <om>
2060 data"im Rheinland angebaut ?"         <op>
2070 data"ja1"                             <pf>
2080 data"nein, werden aus Italien
 importiert "                              <gi>
2090 data"nein, werden aus Holland
 importiert "                              <df>
2100 data"In welchem EG-Land befin
den sich die"                              <gc>
2110 data"groessten Gewaechshausfl
aechen zur      Gemueseproduktion
?"                                         <cb>
2120 data"in der Bundesrepublick D
eutschland "                               <cn>
2130 data"in den Niederlanden2"            <gm>
2140 data"in Italien "                     <cg>
2150 data"Was ist der Unterschied
zwischen"                                  <hn>
2160 data"Zuckermais und Feldmais
?"                                         <hm>
2170 data"es gibt keinen Unterschi
ed "                                       <lm>
2180 data"Feldmais wird auf dem Fe
ld, Zuckermais  wird im Gewaechsha
us angebaut "                              <pl>
2190 data"Zuckermais wird zur Milc
hreife geerntet und schmeckt suess
er3"                                       <ci>
2200 data"Was sind"                        <cn>
2210 data"Kulturraueschlinge ?"            <ln>
2220 data"eine neue Gemueseart "           <bg>
2230 data"eine neue Obstart "              <hm>
2240 data"Pilze, die man auch im e
igenen Garten   anbauen kann3"             <kd>
2250 rem *************** obstbau *         <le>
2260 data"Wie hoch ist der Pro-Kop
f-Verbrauch"                               <lp>
2270 data"an Obst in der Bundesrep
ublik ?"                                   <mk>
2280 data" 60 Kg "                         <an>
2290 data" 80 Kg2"                         <ii>
2300 data"100 Kg "                         <mm>
2310 data"Wo liegen die bedeutenst
en Obst-Anbau-"                            <og>
2320 data"gebiete der Bundesrepubl
ik ?"                                      <pn>
2330 data"in der Lueneburger Heide
 "                                         <ig>
2340 data"im Rheinland2"                   <if>
2350 data"in der Schwaebischen Alp
 "                                         <hg>
2360 data"Wie lautet der Name eine
r fruehen"                                 <oo>
2370 data"heimischen Apfelsorte ?"         <mb>
2380 data"James Grieve "                   <ma>
2390 data"Boskop "                         <ck>
2400 data"Klarapfel3"                      <ka>
2410 data"Welche Apfelsorte enthae
lt"                                        <mc>
2420 data"besonders viel Vitamin C
 ?"                                        <am>
2430 data"Berlepsch1"                      <id>
2440 data"Golden Delicious "               <ja>
2450 data"Cox Orange "                     <jm>
2460 data"Von welcher Obstart esse
n die deutschen"                           <ga>
2470 data"Verbraucher am meisten ?
"                                          <po>
2480 data"Erdbeeren "                      <ok>
2490 data"Aepfel2"                         <co>
2500 data"Apfelsinen "                     <in>
2510 data"Wo liegt das rheinische
Haupt-"                                    <ei>
2520 data"anbaugebiet fuer Aepfel
?"                                         <gk>
2530 data"im Raum Meckenheim1"             <lg>
2540 data"im Raum Kleve "                  <fg>
2550 data"im Grossraum Duesseldorf
```

```
"                                         <jc>
2560 data"Ab welchem Monat gibt es
 Erdbeeren"                               <mj>
2570 data"aus deutschen Landen ?"         <ff>
2580 data"Mai "                           <di>
2590 data"Juni2"                          <ak>
2600 data"Juli "                          <gl>
2610 data" "                              <fe>
2620 data"Erdbeeren wachsen an :"         <gh>
2630 data"Baeumen "                       <kn>
2640 data"Straeuchern "                   <kj>
2650 data"Stauden3"                       <hf>
2660 data"Welche Erdbeersorte laes
st sich"                                  <ho>
2670 data"gut tiefgefrieren ?"            <na>
2680 data"Tenira "                        <fc>
2690 data"Elvira "                        <ob>
2700 data"Senga Sengana3"                 <hh>
2710 data" "                              <hj>
2720 data"Rhabarber ist :"                <dl>
2730 data"eine Obstart "                  <jm>
2740 data"eine Gemueseart2"               <kb>
2750 data"ein Kuechenkraut "              <mb>
2760 rem *** haushalt/verwendung *        <jh>
2770 data"Wieviel Gemuese sollte m
an"                                       <cd>
2780 data"taeglich essen ?"               <bl>
2790 data" 50 - 100 g "                   <li>
2800 data"100 - 150 g "                   <dj>
2810 data"200 - 250 g3"                   <pi>
2820 data"Warum ist frisches Gemue
se fuer die"                              <ci>
2830 data"Ernaehrung so wichtig ?"        <jh>
2840 data"weil es im ganzen Jahr v
erfuegbar ist "                           <fh>
2850 data"weil es viele Vitamine,
Minaeral und    Ballaststoffe enth
aelt2"                                    <fo>
2860 data"weil es so preisguenstig
 ist "                                    <mo>
2870 data"Welche Kraeuter lassen s
ich"                                      <dg>
2880 data"besonders gut trocknen ?
"                                         <ng>
2890 data"Petersilie, Schnittlauch
 "                                        <lc>
2900 data"Bohnenkraut, Majoran, Th
ymian, Rosmarin2"                         <le>
2910 data"Zitronenmelisse, Basilik
um "                                      <ka>
2920 data" "                              <al>
2930 data"Was ist Tripmadam ?"            <lf>
2940 data"ein gaertnerischer Beruf
 "                                        <an>
2950 data"ein Unkraut "                   <cc>
2960 data"ein Kuechenkraut3"              <dk>
2970 data"Welche Teile des Brokkol
is"                                       <mg>
2980 data"sind essbar ?"                  <mp>
2990 data"nur die Blume "                 <ai>
3000 data"die spargelaehnlichen St
iele und die    Blume2"                   <mn>
3010 data"nur die Stiele "                <ho>
3020 data" "                              <cp>
3030 data"Was sind Ballaststoffe ?
"                                         <fp>
3040 data"Zubehoer von Heissluftba
llons "                                   <ed>
3050 data"notwendige Bestandteile
der Ernaehrung2"                          <fc>
3060 data"unnoetige Bestandteile d
er Ernaehrung "                           <ce>
3070 data" "                              <lm>
3080 data"Was ist Nitrat ?"               <eg>
3090 data"ein Salz der Salpetersae
ure und ein     Pflanzennaehrstoff
1"                                        <ae>
3100 data"ein Pflanzengift "              <go>
3110 data"ein Vollduenger "               <hd>
3120 data" "                              <fe>
3130 data"Wie sollte Salat gewasch
en werden ?"                              <bd>
3140 data"mit warmen Wasser "             <ap>
3150 data"mit kaltem Wasser2"             <ab>
3160 data"durch langes waessern "         <cf>
3170 data"Warum sollte Rhabarber i
n den Sommer-"                            <oa>
3180 data"monaten nicht so oft geg
essen werden ?"                           <lb>
3190 data"es ist genug anderes Gem
uese vorhanden "                          <bp>
3200 data"weil er im Sommer nicht
so gut schmeckt "                         <om>
3210 data"weil er im Sommer einen
erhoehten Oxal- saeuregehalt aufwe
ist3"                                     <ec>
3220 data"Woraus sollte eine gesun
de"                                       <nf>
3230 data"Ernaehrung bestehen ?"          <lk>
3240 data"nur aus Gemuese "               <ln>
3250 data"aus gemischter Kost2"           <hg>
3260 data"nur aus Fleisch "               <md>
3270 rem **** biologischer anbau *        <ag>
3280 data""                               <fh>
3290 data"Welche Eigenschaften hab
en Ohrwuermer ?"                          <ai>
3300 data"raeuberische Insekten, d
ie z.b. Laeuse  fressen1"                 <ad>
3310 data"sind schaedlich "               <da>
3320 data"sind weder nuetzliche, n
och schaedlich "                          <jb>
3330 data"Unter welchem Markenzeic
hen werden bio-"                          <ch>
3340 data"logisch-dynamisch erzeug
te Lebensmittel verkauft ?"               <co>
3350 data"ANOG "                          <bn>
```

```
3360 data"Demeter2"                      <mm>
3370 data"Bioland "                      <cn>
3380 data"Wieviel Prozent der land
wirtschaftlich"                          <bl>
3390 data"genutzten Flaeche werden
 alternativ     bewirtschaftet ?"        <gh>
3400 data" 0,2 %1"                       <gi>
3410 data" 1,0 % "                       <cm>
3420 data"10,0 % "                       <ok>
3430 data"Wie wird im biologischen
 Anbau "                                 <eo>
3440 data"geduengt ?"                    <fc>
3450 data"ueberhaupt nicht "             <jp>
3460 data"mit Mineralduengern "          <gi>
3470 data"mit Kompost oder Stallmi
st3"                                     <fj>
3480 data"Wie wird im biologischen
 Anbau"                                  <cn>
3490 data"Pflanzenschutz betrieben
 ?"                                      <ik>
3500 data"ueberhaupt nicht "             <ad>
3510 data"mit ungiftigen Chemische
n Mitteln "                              <an>
3520 data"mit natuerlichen Mitteln
3"                                       <od>
3530 data"Wieviel verschiedene Alt
ernative"                                <bk>
3540 data"Anbaurichtungen gibt es
?"                                       <cc>
3550 data" 2 Anbaurichtungen "           <bd>
3560 data" 5 Anbaurichtungen "           <ab>
3570 data"20 Anbaurichtungen3"           <ee>
3580 data"Welche Massnahme ist bei
 allen alter-"                           <cc>
3590 data"nativen Anbaurichtungen
gleich ?"                                <if>
3600 data"Foerderung der Bodenfruc
htbarkeit1"                              <af>
3610 data"Beachtung von Mond-Rhyth
men "                                    <ae>
3620 data"Spritzen mit Hornmist un
d Hornkiesel "                           <do>
3630 data"Wieviel Insekten kann ei
n Meisenpaar"                            <hp>
3640 data"mit Nachkommen pro Jahr
vertilgen ?"                             <pf>
3650 data"10 Kg "                        <gn>
3660 data"30 Kg "                        <ka>
3670 data"70 Kg3"                        <bl>
3680 data"Welches Insekt ist ein n
atuerlicher"                             <hl>
3690 data"Blattlausfeind ?"              <ip>
3700 data"Maikaefer "                    <ij>
3710 data"Marienkaefer2"                 <lm>
3720 data"Wespe "                        <kg>
3730 data"Welche Gruenduengungspfl
anze kann"                               <pi>
3740 data"Stickstoff sammeln ?"          <de>
3750 data"Lupine1"                       <lj>
3760 data"Senf "                         <hp>
3770 data"Winterroggen "                 <cj>
3780 rem ======================== <ai>
3790 rem 60671 bytes memory      <ei>
3800 rem 11648 bytes program     <kc>
3810 rem 00098 bytes variables   <nm>
3820 rem 00902 bytes arrays      <bl>
3830 rem 06804 bytes strings     <ob>
3840 rem 41219 bytes free (0)    <kh>
3850 rem ======================== <ma>
```

Kasino

Vier Glücksspiele, Color-Set, Change, High-Low und Blue Jack, finden Platz im Speicher des C16/116, und das mit Anleitungen, Sound und guter Grafik. Wieder einmal beweist uns Peter Bergen, was sich aus dem Rechner herausholen läßt.

Als C16-User bekommt man oft zu hören, daß mit nur 12 KByte Speicherplatz kaum ein vernünftiges Programm zu schreiben sei.
Um das Gegenteil zu beweisen, habe ich das Programm Kasino geschrieben. Obwohl man es kaum für möglich hält, befinden sich bei dieser Glücksspielsammlung gleich vier unterschiedliche Programme zur gleichen Zeit im Speicherplatz des C16, wobei auch die Anleitungen mit einprogrammiert sind.
Bei Kasino handelt es sich um Glücksspiele, die in einen Spielautomatenlook verpackt wurden. Alle vier Spiele sind einzeln (gegen den Highscore) oder mit bis zu vier Teilnehmern spielbar.
Die Einsparung an Speicherplatz ist darauf zurückzuführen, daß Ton- und Grafikroutinen gemeinschaftlich von allen vier Spielen genutzt werden; dennoch unterscheiden sich die Spiele deutlich voneinander.

DAS PROGRAMM

Kasino ist so aufgebaut, daß alle vier Spiele einen gemeinsamen Anfangsteil innerhalb der Zeilen 120 bis 220 haben und in Zeile 230 zu den verschiedenen Mittelteilen verzweigen.
Falls nach der Rückkehr ein Gewinn vorliegt, folgt ein gemeinsamer Endteil ab Zeile 240 (Risikomöglichkeit), andernfalls ab Zeile 250.
Innerhalb der zehn Runden wird eine Schleife von Zeile 220 bis 280 gebildet. Nach Ablauf aller Runden erfolgt über die Endroutine (1290 bis 1350) ein Neustart ab Zeile 120.
Die Programmierung wurde nach dem Prinzip konsequenter Unterprogrammierung vorgenommen, so daß es mir sinnvoll erscheint, statt einer konventionellen Programmbeschreibung die Bedeutung der einzelnen Programmabschnitte zu benennen.
Zusammen mit der Variablenliste dürfte so der Algorithmus von Kasino leicht verfolgbar sein.

100– 280 = Hauptprogramm
290– 460 = Mittelteil von COLOR-SET
470– 850 = Mittelteil von CHANCE
860–1010 = Mittelteil von HIGH-LOW
1020–1280 = Mittelteil von BLUE JACK
1290–1350 = Endroutine
1360–1370 = Zufallszahlen K (1–5)
1380–1450 = Zufallsfarben F (1–5)
1460–1470 = Blaufärbung der Karten
1480–1550 = Tasten An/Aus-Schaltung
1560–1590 = Nachziehung der Karten
1600–1740 = Risiko-Routine



Nach Las Vegas in das Spielparadies, fühlt sich der Kasino-Spieler versetzt, das Flamingo wartet auf Sie.

1750–1800 = Punkte- und Highscore-Anrechnung
1810–1880 = Soundroutinen
  1820 = Verlust
  1830 = Gewinn
  1840 = Anfang
  1870 = Ende
1890–2180 = Grafikunterprogramme
  1910 = Mittelfenster
  1920 – 1930 = Spielerfenster
  1940 = Highscorefenster
  1950 – 1960 = Spielerbeschriftung
  1980 = Kartenrückseite
  1990 – 2040 = Kartenvorderseite
  2110 – 2140 = Neue Risikoleiste
  2150 – 2160 = Tasten-An/Aus-Darstellung
  2170 – 2180 = Unterroutine für fünf neue Karten
2190–2330 = Kommentare und Zusatztafeln
  2200 = Leerstring unten
  2210 = Bedienungsstring von Ende
  2220 = Bedienungsstring von COLOR-SET
  2230 = Bedienungsstring von Risiko
  2240 = Rundenhinweis
  2250 = Zusatztafel von CHANCE
  2270 = Bedienungsstring von CHANCE
  2280 = Bedienungsstring von HIGH-LOW
  2290 – 2300 = Hinweis auf Fehlbedienung
  2310 – 2320 = Zusatztafel für BLUE JACK
  2330 = Bedienungsstring für BLUE JACK
2340–2410 = Anfangsmenü
2430 = Anleitungen
  2440–2600 = Anleitung für COLOR-SET
  2610–2810 = Anleitung für CHANCE
  2820–2960 = Anleitung für HIGH-LOW
  2970–3080 = Anleitung für BLUE JACK
  3090–3180 = Anleitung für Risiko
3190 = Pause
3200–3280 = Überschriften
  3210 = COLOR-SET
  3230 = CHANCE
  3250 = HIGH-LOW
  3270 = BLUE JACK
3290 = Universeller Leerstring

VARIABLENLISTE

A,B,C = Universelle FOR-NEXT-Variablen
AG = ASC-Code von G$
AS = Aktueller Spieler
COL = Tastenfarbe
CW = Chance-Gewinnzahlkonto
F = Farbzuweisung
GZ = Risikozufallszahl
GP = Risikobegrenzung
H = Highscore
JP = 1. Spielsumme bei BLUE JACK
K = Kartenwert
KB = Kartenbegrenzung
KP = Aktuelle Karte

Lesen Sie bitte weiter auf Seite 94

```
10 rem kasino==================c16 <np>
20 rem (p) commodore welt <op>
30 rem ========================== <ng>
40 rem (c) by peter bergen <pk>
50 rem <pd>
60 rem <ah>
70 rem basic v3.5 <nl>
80 rem c16/116/plus4 <ki>
90 rem ========================== <jg>
100 gosub 3320 <eo>
110 vol7:color0,1:dimf(5):dimk(5):
dimt(5):dimsk(4):dimct(4):dimh(4) <ln>
120 color1,2:po=12:fora=1to4:sk(a)
=0:nexta:gosub2350 <ao>
130 getkeyg$:ag=asc(g$):scnclr <gg>
140 ifag>52orag<49thengoto130 <kg>
150 scnclr:char1,1,22,"anleitung e
rwuenscht? (j=ja/n=nein)":getkeyx$ <eb>
160 ifx$="n"thengoto180 <ln>
170 onag-48gosub2440,2610,2820,297
0 <cc>
180 color1,2,7:scnclr:char1,1,22,"
":input"wieviel spieler (1-4)";sa <og>
190 ifsa>4thengoto150 <ik>
200 gosub1910:sp=6:as=1:rg=1:rb=10
:b=rnd(-ti) <ed>
210 fora=1to6:sound1,200,2:sound1,
400,2:nexta <ph>
220 p=0:gosub2200:gosub2240:color1
,7,7:gosub1960:gosub1980:gosub2060 <jo>
230 onag-48goto300,480,870,1030 <dg>
240 gosub1610 <nd>
250 gosub1760:color1,7,4:gosub1960 <cj>
260 sp=sp+3:as=as+1:ifas=sa+1thena
s=1:sp=6:rg=rg+1 <dp>
270 ifrg=rb+1 thengoto1300 <mj>
280 goto210 <dj>
290 rem color-set <mn>
300 pr=2:gosub3210:zb=6:kb=6:t1=12
:t2=28:t3=1:gosub2120 <ln>
310 forkp=1to5:gosub1370:gosub1390
:nextkp:gosub2180:ru=1 <ke>
320 gosub2200:gosub2220:gosub1490:
gosub2200:gosub1570:t3=1:gosub2120 <kb>
330 ifru=1thenru=ru+1:goto320 <al>
340 ifg$="3"thenreturn <ff>
350 rem color-set/gewinn <am>
360 a=1 <oc>
370 iff(a)<>2thenmc=f(a):goto390 <pl>
380 a=a+1:ifa<6thengoto370 <ig>
390 a=1 <nf>
400 iff(a)<>mcandf(a)<>2thengoto25
0 <fp>
410 a=a+1:ifa<6thengoto400 <kn>
420 p=k(1)+k(2)+k(3)+k(4)+k(5) <aj>
430 a=1 <bi>
440 iff(a)=2thengoto240 <nn>
450 a=a+1:ifa<6thengoto440 <kk>
460 p=p+p:goto240 <fh>
470 rem chance <ij>
480 pr=2:gosub3230:t1=12:t2=24:t3=
1:gosub2120 <ga>
490 ru=1:zb=5:kb=5:gosub2250:gosub
1470 <dm>
500 forkp=1to5:gosub1370:nextkp:go
sub2180 <oo>
510 gosub2270:gosub1490 <hn>
520 a=1:pr=14 <ja>
530 ift(a)=1andcw(a)>0thengosub229
0:t3=1:gosub2120:goto510 <cc>
540 ift(a)=1andcw(a)=0thenwa=a:got
o570 <eh>
550 a=a+1:pr=pr+5:ifa=5thengosub22
90:goto510 <mk>
560 goto530 <ng>
570 fora=1to5 <bb>
580 ifk(a)=wathencw(wa)=cw(wa)+k(a
) <hh>
590 nexta:color1,6,6:char1,pr,7,rn
$:printusing"##";cw(wa) <cd>
600 ru=ru+1:ifru<5thengosub1980:t3
=1:gosub2120:goto500 <lm>
610 fora=1to4:p=p+cw(a):nexta:colo
r1,6,6:char1,18,8,rn$:printusing"#
#";p <hm>
620 fora=1to4:cw(a)=0:nexta <in>
630 rem zielzahl <ho>
640 zz=int(rnd(1)*21)+10 <kk>
650 fora=1tozz:char1,27,8,rn$:prin
tusing"##";a:sound1,0,1:forb=1to40
:nextb:nexta <in>
660 ifp>=zzthengoto240:elsep=0:got
o250 <ac>
670 rem kartenueberpruefung <fj>
680 fora=1to5 <ln>
690 ift(a)=1thenmc=f(a) <an>
700 nexta <ea>
710 a=1 <np>
720 ift(a)=1andmc<>f(a)thensound1,
2,5:gosub1490:goto680 <jj>
730 a=a+1:ifa=6thenep=7:goto750 <pl>
740 goto720 <cd>
750 fora=1to5:ift(a)=1thenp1=p1+k(
a) <hb>
760 nexta:color1,6,6:char1,18,7,rn
$:printusing"##";p1:return <fb>
770 fora=1to5 <fe>
780 ift(a)=1thenmc=k(a) <gl>
790 nexta <kk>
800 a=1 <le>
810 ift(a)=1andmc<>k(a)thensound1,
2,5:gosub1490:goto770 <ge>
820 a=a+1:ifa=6thenep=7:goto840 <nb>
830 goto810 <ml>
840 fora=1to5:ift(a)=1thenp2=p2+k(
a) <ml>
```

```
850 nexta:color1,6,6:char1,18,8,rn
$:printusing"##";p2:return          <lm>
860 rem high-low                    <gp>
870 pr=2:gosub3250:t1=16:t2=28:t3=
2:gosub2120                         <hk>
880 zb=10:gosub1470:po=12:forkp=1t
o5:gosub1370:nextkp:kp=1            <oi>
890 gosub1990                       <ij>
900 gosub2280:gosub960              <kb>
910 ift(kp)=1andk(kp-1)<=k(kp)then
goto940                             <jd>
920 ift(kp)=0andk(kp-1)>=k(kp)then
goto940                             <dc>
930 gosub1990:goto250               <nk>
940 ifkp=5thengosub1990:fora=1to5:
p=p+k(a):nexta:goto240              <ce>
950 goto890                         <gh>
960 po=po+4:kp=kp+1                 <gp>
970 getkeyx$                        <ji>
980 ifasc(x$)-48=kpandt(kp)=1thent
(kp)=0:col=3:gosub2160:goto970      <pa>
990 ifasc(x$)-48=kpandt(kp)=0thent
(kp)=1:col=7:gosub2160:goto970      <hi>
1000 ifasc(x$)=13thenreturn         <oe>
1010 goto970                        <cc>
1020 rem blue-jack                  <ng>
1030 pr=2:gosub3270                 <pi>
1040 pj=0:en=0:pr=7:gosub2200:char
1,1,1,rn$:gosub2310:char1,1,1,rf$:
gosub2320                           <mk>
1050 zb=10:gosub1470:forkp=1to5:go
sub1370:nextkp:kp=1:po=12           <kj>
1060 gosub1990                      <ok>
1070 p=p+k(kp):char1,1,1,rn$:gosub
1280                                <db>
1080 gosub2330:po=po+4:kp=kp+1      <np>
1090 getkeyx$                       <cb>
1100 ifasc(x$)=13thengoto1130       <pm>
1110 ifasc(x$)=19thengoto1180       <gn>
1120 goto1090                       <ab>
1130 gosub1990                      <jp>
1140 p=p+k(kp):char1,1,1,rn$:gosub
1280                                <ek>
1150 ifp>17thenp=0:goto250          <nl>
1160 ifkp=5thengoto1180             <pl>
1170 goto1080                       <fe>
1180 gosub2200                      <nh>
1190 char1,1,1,rf$:gosub1980:gosub
2310:gosub1280:pr=8:char1,1,1,rn$:
gosub2320:jp=p:p=0                  <ch>
1200 zb=10:gosub1470:forkp=1to5:go
sub1370:nextkp:kp=1:po=12           <lj>
1210 gosub1990:p=p+k(kp):char1,1,1
,rn$:gosub1280                      <ng>
1220 ifp>17thengoto240              <la>
1230 ifp>=jpthenp=0:goto250         <dp>
1240 ifkp=5thengoto240              <cg>
1250 po=po+4:kp=kp+1                <ja>
1260 fora=1to500:nexta:goto1210     <oo>
1270 ifg$="4"thenp=p+k(kp):char1,1
,1,rn$:gosub1280                    <fc>
1280 color1,6,6:char1,26,pr,"":pri
ntusing"##";p:return                <jj>
1290 rem ende                       <al>
1300 gosub2060:color1,2,7:char1,13
,12,"":print"e"c3$c3$c3$"n";        <ha>
1310 printc3$c3$c3$"d"c3$c3$c3$"e"
c3$c3$c3$"!":gosub1870:gosub2200:g
osub2210                            <hj>
1320 getkeyx$                       <jg>
1330 ifx$="j"thengoto120            <no>
1340 ifx$="n"thenstop               <fc>
1350 goto1320                       <ni>
1360 rem zufallszahlen              <fn>
1370 k(kp)=int(rnd(1)*zb):return    <mg>
1380 rem zufallsfarben              <fm>
1390 col=int(rnd(1)*4)+1            <ai>
1400 ifk(kp)=0thenf(kp)=2:return    <dn>
1410 ifcol=1thenf(kp)=3:return      <eo>
1420 ifcol=2thenf(kp)=6:return      <kn>
1430 ifcol=3thenf(kp)=7:return      <ll>
1440 ifcol=4thenf(kp)=8:return      <kj>
1450 return                         <on>
1460 rem blaue karten               <kg>
1470 fora=1to5:f(a)=7:nexta:return  <ja>
1480 rem tasten an/aus+start        <jn>
1490 getkeyx$                       <dl>
1500 kp=1:po=12                     <kp>
1510 ifasc(x$)-48=kpandt(kp)=1then
t(kp)=0:col=3:gosub2160:goto1490    <eg>
1520 ifasc(x$)-48=kpandt(kp)=0then
t(kp)=1:col=7:gosub2160:goto1490    <kf>
1530 kp=kp+1:po=po+4:ifkp<kbthengo
to1510                              <al>
1540 ifasc(x$)=13thenreturn         <nb>
1550 goto1490                       <bh>
1560 rem nachziehung                <lo>
1570 po=12:forkp=1to5               <bn>
1580 ift(kp)=1thengosub1370:gosub1
390:gosub1990                       <gi>
1590 po=po+4:nextkp:return          <ig>
1600 rem risiko                     <ec>
1610 so=200:pr=p:color1,8,7:fora=1
6to6step-2                          <eo>
1620 char1,34,a,rn$:printusing"###
#";pr                               <ff>
1630 pr=pr*2:nexta:gosub2080        <pp>
1640 gp=15:gosub2090                <ii>
1650 gosub2230                      <ma>
1660 ifgp=5thenreturn               <ef>
1670 gosub2090:gz=int(rnd(1)*2)+1   <bc>
1680 getx$                          <ap>
1690 sound1,so,10                   <ea>
1700 ifasc(x$)=19thengosub1830:ret
urn                                 <eh>
1710 ifasc(x$)=13andgz=1thenp=p*2:
```

```
gp=gp-2:so=so+100:goto1660          <lb>
1720 ifasc(x$)=13andgz=2thenp=0:go
sub1820:return                      <ap>
1730 sound1,so+150,10               <ah>
1740 goto1680                       <of>
1750 rem punkteanrechnung           <pi>
1760 color1,2,6:sk(as)=sk(as)+p:ch
ar1,2,sp+1,rn$:printusing"#######"
;sk(as)                             <op>
1770 fora=1to4                      <ch>
1780 ifsk(a)>h(ag-48)thenh(ag-48)=
sk(a)                               <kc>
1790 nexta                          <fg>
1800 color1,2,6:char1,4,19,rn$:pri
ntusing"#####";h(ag-48):return      <ck>
1810 rem sound                      <bj>
1820 fora=soto0step-10:sound1,a,2:
sound2,a+100,2:nexta:fora=1to500:n
exta:return                         <kp>
1830 fora=1tosostep20:sound1,a,5:s
ound2,so,2:nexta:fora=1to500:nexta
:return                             <jo>
1840 restore1850:fora=1to16:readb,
c:sound1,b,c:nexta:return           <gh>
1850 data810,30,784,8,770,20,704,2
0,685,20,739,20,704,20,770,20,739,
20,784,20                           <if>
1860 data770,20,810,20,596,20,685,
20,704,20,704,20                    <hm>
1870 restore1880:fora=1to7:readb,c
:sound1,b,c:nexta:return            <ac>
1880 data810,30,834,8,810,8,784,8,
770,20,704,20,704,20                <hc>
1890 rem grafikunterprogramme       <kd>
1900 rem grundaufbau                <hf>
1910 scnclr:color1,2,7:fora=6to16:
char1,11,a,rn$+b$+b$+" ":nexta      <ja>
1920 sp=6:foras=1tosa:color1,7,4:g
osub1960:color1,2,6:char1,1,sp+1,r
n$+b5$+b4$+rf$                      <nc>
1930 sp=sp+3:nextas                 <an>
1940 color1,7,6: char1,1,18,rn$+"h
ighscore":color1,2,6:char1,1,19,b5
$+b4$+rf$:return                    <ob>
1950 rem spielermarkierung          <lb>
1960 char1,1,sp,rn$:printusing"#";
as:char1,2,sp,rn$+".spieler"+rf$:r
eturn                               <bk>
1970 rem kartendarstellung          <kk>
1980 color1,2,5:fora=12to30step4:f
orb=10to14:char1,a,b,z5$+z5$+z5$:n
extb:nexta:return                   <oo>
1990 color1,2,7:char1,po,10,rn$+b3
$:char1,po,14,b3$:gosub3190         <cd>
2000 char1,po,11,b3$:char1,po,13,b
3$:gosub3190                        <ed>
2010 char1,po,12,zv$+zv$+zv$:gosub
3190:char1,po,12,b3$                <di>
2020 color1,f(kp),5:char1,po,12,rn
$+" - ":gosub3190                   <ff>
2030 char1,po,1[illegible]3$:char1,po,13,b
3$:gosub3190                        <lp>
2040 char1,po,1[illegible]3$:char1,po,14,b
3$:printc1$c1$c1$c2$c2$;k(kp):soun
d1,10,.2:return                     <be>
2050 rem gamblingneuanfang          <kf>
2060 fora=6to16step2:color1,8,7:ch
ar1,34,a,rn$+b4$+rf$:color1,3,4:ch
ar1,35,a+1,"↑↑"                     <gf>
2070 nexta:char1,35,17,b2$:char1,3
4,18,"0000":return                  <bh>
2080 color1,3,6:char1,34,18,f1$+"0
000"+fo$:return                     <ee>
2090 color1,3,6:char1,35,gp,rf$+fl
$+"↑↑"+fo$:color1,3,4:char1,35,gp+
2,"↑↑"                              <lo>
2100 char1,35,17,b2$:return         <ml>
2110 rem tastendarstellung          <jd>
2120 color1,3,3:fora=t1tot2step4:f
orb=18to20:char1,a,b,rn$+b3$        <po>
2130 nextb:printc1$c1$c1$c2$;t3:t3
=t3+1:nexta                         <cb>
2140 fora=1to5:t(a)=0:nexta:return  <cb>
2150 rem tasten an/aus              <ln>
2160 color1,3,col:forb=18to20:char
1,po,b,rn$+b3$:nextb:printc2$c1$c1
$c1$;kp:return                      <oi>
2170 rem 5 karten                   <kn>
2180 po=12:forkp=1to5:gosub1990:po
=po+4:nextkp:return                 <be>
2190 rem kommentare und tafeln      <fj>
2200 color1,2,6:char1,1,22,rn$+b$+
b$+b$+b4$+b4$+rf$:return            <ad>
2210 color1,2,6:char1,2,22,rn$+"no
ch ein spiel? (j=ja / n=nein)"+rf$
:return                             <ke>
2220 color1,2,6:char1,2,22,rn$+"1-
5 = an/aus, return = start"+rf$:re
turn                                <pp>
2230 color1,2,6:char1,2,22,rn$+"cl
ear home =annahme, return=risiko":
return                              <mk>
2240 color1,2,6:char1,2,22,rn$:pri
nt"runde";rg;" von";rb;"runden"rf$
:return                             <hl>
2250 color1,6,6:char1,12,7,rn$+"1:
00":char1,17,7,"2:00":char1,22,7,"
3:00"                               <kn>
2260 char1,27,7,"4:00":char1,12,8,
"summe:"+b3$+"ziel :"+b4$+rf$:retu
rn                                  <ai>
2270 color1,2,6:char1,2,22,rn$+"1-
4 = chance-wahl, return=start"+rf$
:return                             <gi>
2280 color1,2,6:char1,2,22,rn$:pri
nt"taste ";kp+1;"an=>, aus=<, retu
```

```
rn=start":return                                <oc>
2290 color1,2,6:char1,2,22,rn$+"bi
tte waehlen sie noch einmal!"+rf$    <ne>
2300 sound1,0,50:fora=1to800:nexta
:return                                         <cc>
2310 color1,6,6:char1,12,7," ihre"
+b2$+"summe:"+b6$:return                        <la>
2320 color1,6,6:char1,12,8," jacks
 summe:"+b6$:return                             <bo>
2330 color1,2,6:char1,1,22,rn$+"re
turn=naechste karte, clr home=genu
g"+rf$:return                                   <lg>
2340 rem anfangsmenue                           <am>
2350 scnclr:color4,1:b=1:color1,2,
7:fora=28to6step-1:char1,a,b,rn$+"
kasino"+rf$:b=b+1                               <fb>
2360 nexta:gosub1840                            <gb>
2370 pr=11:gosub3290:pr=12:gosub32
10:pr=13:gosub3290:pr=14:gosub3230
:pr=15:gosub3290                                <jm>
2380 pr=16:gosub3250:pr=17:gosub32
90:pr=18:gosub3270:pr=19:gosub3290  <hj>
2390 color1,2,7:char1,2,1,"bitte w
aehlen sie:"                                    <pg>
2400 char1,1,12,"1 = ":char1,1,14,
"2 = ":char1,1,16,"3 = ":char1,1,1
8,"4 = "                                        <kl>
2410 return                                     <he>
2420 goto2420                                   <co>
2430 rem anleitungen                            <ij>
2440 scnclr:color1,3,7:color4,3,6  <ec>
2450 printchr$(14)                              <bj>
2460 printc4$c4$c3$"Innerhalb drei
er Ziehungen sollen Sie"                        <mh>
2470 print"erreichen, dass alle 5
Karten die selbe"                               <gl>
2480 print"Farbe vorweisen."                    <ek>
2490 print"Die Karten haben Werte
von 0-5 und die"                                <ga>
2500 print"Farben: Gelb,Blau,Gruen
,Rot und Grau."                                 <ib>
2510 print"Graue Karten mit >0< ge
lten als Ersatz"                                <pl>
2520 print"fuer fehlende Farben."  <lg>
2530 print"Zwischen jeder Ziehung
koennen Sie mit"                                <bo>
2540 print"den Tasten 1-5 entschei
den, welche Karte"                              <ak>
2550 print"veraendert werden soll,
 wonach die Zieh-"                              <mo>
2560 print"ung mit >RETURN< ausgel
oest wird."                                     <cb>
2570 print"Bei Erreichen des Spiel
ziels gilt die"                                 <ic>
2580 print"Summe der 5 Kartenwerte
 als Gewinn."                                   <fe>
2590 print"Sollte keine >0< dabeis
ein, wird Ihr"                                  <mp>
2600 print"Gewinn verdoppelt.":get
keyx$:goto3090                                  <ph>
2610 scnclr:color1,8,7:color4,8,6  <hl>
2620 printchr$(14)                              <ki>
2630 printc4$c4$c3$"Die Kartenwert
e von 1-4 sollen in"                            <ah>
2640 print"4 Ziehungen den Gewinnz
ahlen oberhalb"                                 <lp>
2650 print"der Karten zugeordnet w
erden."                                         <ap>
2660 print"Hierzu koennen Sie sich
 nach jeder Zieh-"                              <dm>
2670 print"ung aller 5 Karten eine
 Gewinnzahl durch"                              <am>
2680 print"die Tasten 1-4 auswaehl
en."                                            <ch>
2690 print"Nach >RETURN< wird jede
r Kartenwert der"                               <bf>
2700 print"dieser Gewinnzahl entsp
richt auf das ne-"                              <ih>
2710 print"benstehende Konto der G
ewinnzahl ge-."                                 <ha>
2720 print"bucht. Es duerfen nur G
ewinnzahlen ge-"                                <nb>
2730 print"waehlt werden, deren Ko
nto noch auf >00<"                              <fo>
2740 print"steht."                              <mm>
2750 print"Nach Ablauf der 4 Ziehu
ngen bilden alle"                               <ph>
2760 print"Konten eine Summe, die
mit einer Zu-"                                  <fe>
2770 print"fallszahl zwischen 10 u
nd 30; dem sog."                                <ah>
2780 print">Ziel< verglichen wird.
 Ist das Ziel"                                  <kl>
2790 print"niedriger oder gleich,
so gilt die"                                    <mj>
2800 print"Summe als gewonnen."    <ld>
2810 getkeyx$:goto3090                          <an>
2820 scnclr:color1,6,7:color4,6,6  <bd>
2830 printchr$(14)                              <hl>
2840 printc4$c4$c3$"Es finden 5 Zi
ehungen einzelner Karten"                       <pj>
2850 print"mit den Werten 0-9 stat
t. Nach jeder"                                  <id>
2860 print"Ziehung muessen Sie mit
 der angezeigten"                               <db>
2870 print"Taste entscheiden, ob d
er nachfolgende"                                <ep>
2880 print"Kartenwert hoeher (Tast
e an), oder nie-"                               <ob>
2890 print"driger (Taste aus) als
der vorhergehende"                              <dl>
2900 print"Kartenwert ist."                     <ci>
2910 print"Sollte Ihr Tip falsch s
ein, gilt das"                                  <hh>
2920 print"Spiel als verloren. Bei
 Gleichstand wird"                              <oi>
```

```
2930 print"zu Ihren Gunsten entsch
ieden."                              <pi>
2940 print"Bei richtiger Entscheid
ung bilden die"                      <ap>
2950 print"Kartenwerte eine Gewinn
summe."                              <ml>
2960 getkeyx$:goto3090               <jc>
2970 scnclr:color1,7,7:color4,7,6    <dm>
2980 printchr$( 14)                  <cc>
2990 printc4$c4$c3$"Versuchen Sie
durch Nachziehen (RETURN"            <lm>
3000 print"druecken) von max.4 Kar
ten der Werte 0-9"                   <eg>
3010 print"so nah wie moeglich an
die Zahl >17< zu"                    <di>
3020 print"kommen, ohne diesen Wer
t zu uebersteі-"                     <hl>
3030 print"gen."                     <kj>
3040 print"Mit der >CLEAR-HOME<-Ta
ste koennen Sie"                     <dj>
3050 print"Ihr Spiel abbrechen, wo
nach >JACK< gegen"                   <ei>
3060 print"Sie spielt. Sie haben g
ewonnen, wenn"                       <jb>
3070 print"Ihre Spielsumme 17 nich
t uebersteigt und"                   <ld>
3080 print"hoeher als >JACKS< Summ
e ist.":getkeyx$                     <il>
3090 printchr$( 14)                  <lg>
3100 printc4$"Sie haben danach die
 Moeglichkeit Ihren"                 <cj>
3110 print"Gewinn durch die Risiko
funktion zu ris-"                    <fg>
3120 print"kieren oder zu vervielf
achen. Hierbei"                      <lk>
3130 print"koennen Sie die Zahl un
terhalb des roten"                   <fi>
3140 print"Pfeils annemen (CLEAR-H
OME-Taste), oder"                    <ln>
3150 print"die Zahl ueber dem Pfei
l mit Risiko (RE-"                   <ah>
3160 print"TURN-Taste) gewinnen."    <ff>
3170 printc4$"Das Spiel endet nach
 10 Runden."                         <md>
3180 getkeyx$:printchr$( 142):retur
n                                    <db>
3190 fort=1to30:nextt:return         <dj>
3200 rem ueberschriften              <an>
3210 color4,3,6:color1,3,6:char1,1
,pr,rn$+"************* color-set *
*************"+rf$                   <ff>
3220 return                          <mp>
3230 color4,8,6:color1,8,6:char1,1
,pr,rn$+"############### chance ##
#############"+rf$                   <gp>
3240 return                          <ph>
3250 color4,6,6:color1,6,6:char1,1
,pr,rn$+"+++++++++++++ high-low +
+++++++++++++"+rf$                   <io>
3260 return                          <bp>
3270 color4,7,6:color1,7,6:char1,1
,pr,rn$+"$$$$$$$$$$$$$ blue-jack $
$$$$$$$$$$$$$"+rf$                   <bd>
3280 return                          <eh>
3290 char1,1,pr,b$+b$+b$+b6$:retur
n                                    <do>
3300 rem nachspann ============== <ei>
3310 rem * farbcodes/steuercodes * <ja>
3320 c4$=chr$( 017):rn$=chr$( 018)   <gh>
3330 c3$=chr$( 029):f1$=chr$( 130)   <ln>
3340 fo$=chr$( 132):c2$=chr$( 145)   <ac>
3350 rf$=chr$( 146):c1$=chr$( 157)   <bc>
3360 rem *** zeichensatz/graphik * <fg>
3370 z5$=chr$( 166):zv$=chr$( 192)   <dk>
3380 b$=chr$( 32):b2$=b$+b$          <hi>
3390 b3$=b2$+b$:b4$=b3$+b$           <of>
3400 b$=b4$+b4$+b2$:return           <ca>
3410 rem ======================== <pk>
3420 rem 12277 bytes memory          <le>
3430 rem 10927 bytes program         <al>
3440 rem 00245 bytes variables       <kn>
3450 rem 00207 bytes arrays          <lf>
3460 rem 00378 bytes strings         <ki>
3470 rem 00520 bytes free (0)        <lj>
3480 rem ======================== <kp>
```

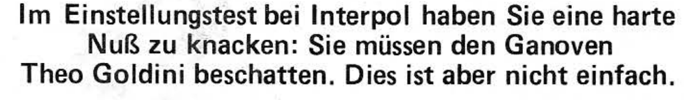
Kasino

Fortsetzung von Seite 89

MC	=	Farbmarkierung
P	=	Aktuelle Punktzahl
P1	=	Zwischensumme bei CHANCE
P2	=	Zwischensumme bei CHANCE
PJ	=	Punktezahl für JACK bei BLUE JACK
PO	=	Kartenposition
PR	=	Universelle Printposition
RB	=	Rundenbegrenzung (steht auf 10, kann beliebig verändert werden)
RG	=	Große Runde
RP	=	Risikoposition
RU	=	Kleine Runde
SA	=	Spieleranzahl
SK	=	Spielerkonto
SO	=	Sound-Höhenfestlegung
SP	=	Spieleranzeigeposition
T	=	Tastennummer
T1	=	1. Schleifenvariable für Tastendarstellung
T2	=	2. Schleifenvariable für Tastendarstellung
T3	=	Tatenaufschrift
WA	=	Wahl bei CHANCE
ZB	=	Zufallsbegrenzung
ZZ	=	Zielzahl bei CHANCE
G$	=	Spielwahl-KEY$
X$	=	Universeller KEY$

□

interpol



Im Einstellungstest bei Interpol haben Sie eine harte Nuß zu knacken: Sie müssen den Ganoven Theo Goldini beschatten. Dies ist aber nicht einfach.

Zu jedem Bild dieses Text-Adventures ist die richtige Antwort, bestehend aus den Buchstaben A, B oder C, zu finden. Eine einzige falsche Antwort wirft Sie unweigerlich aus dem Rennen. Probieren Sie es von Neuem, so müssen Sie plötzlich feststellen, daß jetzt andere Antworten richtig sind.
Glücklicherweise herrscht hier nicht blindlings der Zufall, sondern ein sich nach einer gewissen Anzahl von Spielen ständig wiederholendes Schema.



1:	2:	3:	4:	5:	6:	7:	8:	9:	10:	11:	12:

Ein Tip sei Ihnen zur Lösungsfindung noch gegeben: Nach je fünf Spielen wiederholt sich das Schema. Es gilt also nur noch die richtigen Lösungsbuchstaben herauszufinden und in die obige Tabelle einzutragen. Haben Sie diese kriminalistische Kleinarbeit vollbracht, sind Sie ein Meisterdetektiv, dem kein Theo Goldini mehr etwas vormachen kann. □

```
10 rem interpol================c16 <eb>
20 rem (p) commodore welt team <ho>
30 rem ========================== <ng>
40 rem (c) by peter katzemich <kf>
50 rem          berg. gladbach <kh>
60 rem <ah>
70 rem basic v3.5 <nl>
80 rem c16/116/plus4 <ki>
90 rem ========================== <jg>
100 gosub 2210 <ea>
110 printchr$(14):poke1351,128:e$=
d$+fl$+">>> PRESS"+s2$+" SPACE <<<
"+fo$ <lm>
120 aa$=cl$+rn$+"Loesungszahl:":b$
=rf$+d$+rn$+"Punkte:":l=0:pu=0:be=
25000:lv=50000 <ba>
130 color4,1:color0,1:color1,2 <mf>
140 printaa$;l;b$;pu:char1,11,6,"*
******************" <ic>
150 char1,11,7,"*"+d$+b4$+b3$+"*" <gh>
160 char1,11,8,"* I N T"+s2$+"E"+s
2$+"R"+s2$+"P"+s2$+"O"+s2$+"L *" <ea>
170 char1,11,9,"*"+d$+b4$+b3$+"*":
char1,11,10,"********************" <ma>
180 char1,17,14,"WRITTEN":char1,20
,16,"BY":char1,14,18,"H.P. KATZEMI
CH" <lg>
190 char1,11,23,fl$+">>> PRESS"+s2
$+s2$+"SPACE <<<"+fo$:getkeya$:ifa
$=" "then200:else190 <fn>
200 printcl$c4$c4$c4$" "rn$" I"s2$
"N"s2$"T"s2$"E"s2$"R"s2$"P"s2$"O"s
2$"L : "rf$ <fd>
210 printc4$c4$" Sie haben sich be
i Interpol um":printc4$" einen Arb
eitsplatz beworben." <cb>
220 printc4$" als 'Einstellungstes
t' muessen Sie" <bj>
230 printc4$" den Ganoven Theo Gol
dini beschatten." <jp>
240 printc4$" Dies ist aber nicht
einfach!" <pd>
250 printc4$c4$" Viel Glueck!!!"le
ft$(qd$,4):printe$ <po>
260 getkeya$:ifa$=" "then280:else2
60 <kg>
270 restore <kk>
280 readl:ifl=0then270:else300 <ja>
290 data1,2,3,4,5,0 <mh>
300 color4,6:color0,6:color1,1 <jc>
310 printaa$;l;b$;pu <jj>
320 printleft$(qd$,4)" Theo Goldin
i faehrt mit dem Fahrrad" <jd>
330 printc4$" eine Strasse in Neap
el entlang." <nl>
340 printc4$c4$c4$" "rn$" Wie woll
en sie Theo verfolgen? "rf$ <fa>
350 printc4$b5$"A. mit Ihrem priva
ten PKW" <oo>
360 printc4$b5$"B. mit dem Polizei
 PKW":printc4$b5$"C. mit dem Fahrr
ad" <am>
370 getkeya$:ifa$="a"ora$="b"ora$=
"c"then380:else370 <ho>
380 ifa$=mid$("acabc",l,1)then420 <pb>
390 scnclr:char1,1,10,"Sie haben s
ich zu auffaellig" <eb>
400 char1,1,12,"benommen. Theo hat
 Sie entdeckt und":char1,1,14,"abg
eschuettelt." <mp>
410 goto2120 <pg>
420 color4,1:color0,1:color1,6:pu=
100:printaa$;l;b$;pu <cm>
430 printc4$c4$c4$" Theo trifft si
ch mit einigen" <cd>
440 printc4$" Ganoven in einer Kne
ipe, um einen" <km>
450 printc4$" Bankraub vorzubereit
en.":printc4$c4$" "rn$" Wie verhal
ten Sie sich? "rf$ <gg>
460 printc4$b5$"A. gehen Sie in di
e Kneipe, um den":printc4$b4$b4$"P
lan zu erfahren" <bf>
470 printc4$b5$"B. alamieren Sie d
ie Polizei, um" <lp>
480 printc4$b4$b4$"die Ganoven zu
verhaften" <mj>
490 printc4$b5$"C. Sie warten drau
ssen" <dc>
500 getkeya$:ifa$="a"ora$="b"ora$=
"c"then510:else500 <nn>
510 ifa$=mid$("cbcab",l,1)then540 <on>
520 scnclr:char1,1,12,"Theo ist mi
t den Ganoven durch" <oc>
530 char1,1,14,"die Hintertuer der
 Kneipe abgehauen.":goto2120 <ok>
540 color4,6:color0,6:color1,1:pu=
pu+150:printaa$;l;b$;pu <ef>
550 printc4$c4$c4$" Theo und die G
anoven verlassen die" <hj>
560 printc4$" Kneipe. Sie gehen ih
nen nach. In" <ea>
570 printc4$" einer dunklen Gasse
kommen die Ganoven" <dp>
580 printc4$" auf Sie zu und bedro
hen Sie.":printc4$c4$" "rn$" Was m
achen Sie nun? "rf$ <ji>
590 printc4$b5$"A. Sie erschiessen
 die Ganoven" <bo>
600 printc4$b5$"B. Sie verlassen s
ich auf Ihre":printc4$b4$b4$"Karat
e Kenntnisse" <bd>
610 printc4$b5$"C. Sie laufen schn
ell weg" <km>
620 getkeya$:ifa$="a"ora$="b"ora$=
"c"then630:else620 <jl>
```

```
630 ifa$=mid$("babba",l,1)then700   <gl>
640 ifa$="c"then680                 <id>
650 scnclr:char1,1,10,"Ihr Plan di
e Ganoven zu erledigen"             <if>
660 char1,1,12,"ist nicht geglueck
t. Die Ganoven"                     <ie>
670 char1,1,14,"haben Sie erschoss
en.":goto1820                       <jc>
680 scnclr:char1,1,12,"Die Ganoven
 haben Sie in den Ruecken":char1,1
,14,"geschossen."                   <ab>
690 goto1820                        <cf>
700 color4,1:color0,1:color1,6:pu=
pu+255:printaa$;l;b$;pu             <cf>
710 printc4$c4$c4$" Durch die Ause
inandersetzung mit"                 <pf>
720 printc4$" den Ganoven haben Si
e Theo"                             <co>
730 printc4$" aus den Augen verlor
en.":printc4$c4$" "rn$;             <mn>
740 print" Wo werden Sie Theo nun
suchen? "rf$                        <cp>
750 printc4$b5$"A. In der Kneipe":
printc4$b4$b4$"'ZUM"s2$"SCHRAEGEN"
s2$"VOGEL'"                         <go>
760 printc4$b5$"B. In der Bar am B
ahnhof":printc4$b5$"C. In Theo's W
ohnung"                             <gc>
770 printc4$b5$"D. Im Spielcasino
'ROYAL'"                            <lg>
780 getkeya$:ifa$="a"ora$="b"ora$=
"c"ora$="d"then790:else780          <cb>
790 ifa$=mid$("abdcd",l,1)then810   <fk>
800 scnclr:char1,1,12,"Sie haben T
heo Goldini nicht gefunden.":goto2
120                                 <bg>
810 color4,6:color0,6:color1,1:pu=
pu+245:printaa$;l;b$;pu             <dp>
820 printc4$c4$c4$" Sie haben Theo
 nach langer Sucherei"              <jm>
830 printc4$" wiedergefunden. Dadu
rch sind Sie sehr"                  <cj>
840 printc4$" muede und muessen si
ch einen Tag":printc4$" ausruhen."  <jc>
850 printc4$c4$" "rn$" Wem erteile
n Sie den Auftrag, Theo "rf$        <dh>
860 printc4$" "rn$" fuer einen Tag
 zu beschatten? "rf$                <af>
870 printc4$b5$"A. Ihrem Freund":p
rintc4$b5$"B. Einem Privatdedektiv
"                                   <fe>
880 printc4$b5$"C. Einem Poliziste
n":printc4$b5$"D. Ihrem Bruder"     <bh>
890 getkeya$:ifa$="a"ora$="b"ora$=
"c"ora$="d"then900:else890          <na>
900 ifa$=mid$("cabcd",l,1)then930   <da>
910 scnclr:char1,1,11,"Theo Goldin
i hat Ihren Kollegen"               <oe>
920 char1,1,13,"abgeschuettelt.":g
oto2120                             <ng>
930 color4,1:color0,1:color1,6:pu=
pu+250:printaa$;l;b$;pu             <km>
940 printc4$" Am nechsten Morgen f
olgen Sie Theo"                     <kh>
950 printc4$" auf den Flughafen vo
n Neapel."                          <dj>
960 printc4$" Dort kauft er ein Ti
cket nach Paris"                    <eb>
970 printc4$" von 2000 DM. Leider
haben Sie"                          <lg>
980 printc4$" nur noch 900 DM zur
Verfuegung!"                        <ih>
990 printc4$c4$" "rn$" Wie verhalt
en Sie sich? "rf$                   <df>
1000 printc4$b5$"A. Sie fahren mit
 dem Auto"                          <lm>
1010 printc4$b4$b4$"nach Paris":pr
intc4$b5$"B. Sie fahren mit dem Zu
g"                                  <gm>
1020 printc4$b5$"C. Sie verlangen
einen Gehalts-":printc4$b4$b4$"vor
schuss"                             <gp>
1030 getkeya$:ifa$="a"ora$="b"ora$
="c"then1040:else1030               <gc>
1040 ifa$=mid$("bccaa",l,1)then107
0                                   <dm>
1050 scnclr:char1,1,11,"Am Flughaf
en ist Ihnen ihre Geldboerse"       <ji>
1060 char1,1,13,"gestohlen worden.
":goto2120                          <ki>
1070 color4,6:color0,6:color1,1:pu
=pu+155:printaa$;l;b$;pu            <lk>
1080 printc4$c4$c4$" Theo trifft s
ich in Paris mit einigen"           <bf>
1090 printc4$" Ganoven. Dort wolle
n sie eine Bank":printc4$" ausraub
en."                                <om>
1100 printc4$c4$" "rn$"Welche Moeg
lichkeit werden Sie waehlen?"rf$    <hi>
1110 printc4$b5$"A. Sie benachrich
tigen die Polizei"                  <nm>
1120 printc4$b5$"B. Sie benachrich
tigen die Bank"                     <kj>
1130 printc4$b5$"C. Sie versuchen
Theo auf frischer":printc4$b4$b4$"
Tat zu ertappen"                    <kp>
1140 getkeya$:ifa$="a"ora$="b"ora$
="c"then1150:else1140               <hg>
1150 ifa$=mid$("ababb",l,1)then118
0                                   <ni>
1160 scnclr:char1,1,11,"Bei einem
Schusswechsel sind Sie"             <eg>
1170 char1,1,13,"getoetet worden."
:goto1820                           <fl>
1180 color4,1:color0,1:color1,6:pu
=pu+545:printaa$;l;b$;pu            <db>
```

```
1190 printc4$c4$c4$" Theo flieht a
uf ein Rundfahrten"                  <fj>
1200 printc4$" Schiff. Sie kommen
leider zu"                           <ha>
1210 printc4$" spaet, das Schiff f
aehrt schon auf":printc4$" der Sei
ne."                                 <bg>
1220 printc4$c4$" "rn$" Was werden
 Sie jetzt unternehmen? "rf$         <pn>
1230 printc4$b5$"A. Sie fahren mit
 dem Auto nach"                      <fo>
1240 printc4$b5$"B. Sie warten auf
 das nechste":printc4$b4$b4$"Schif
f"                                   <ea>
1250 printc4$b5$"C. Sie fliegen mi
t dem Polizei":printc4$b4$b4$"Heli
kopter hinterher"                    <af>
1260 getkeya$:ifa$="a"ora$="b"ora$
="c"then1270:else1260                <hg>
1270 ifa$=mid$("bcbac",1,1)then129
0                                    <ba>
1280 scnclr:char1,1,12,"Sie haben
das falsche Schiff verfolgt.":goto
2120                                 <ei>
1290 color4,6:color0,6:color1,1:pu
=pu+355:printaa$;l;b$;pu             <hp>
1300 printc4$c4$c4$" Theo's Spur f
uehrt vom Schiff zu"                 <mp>
1310 printc4$" einer grossen Villa
. Die Villa ist"                     <lk>
1320 printc4$" umgeben von einer h
ohen Mauer."                         <hk>
1330 printc4$c4$" "rn$" Als was we
rden Sie sich verkleiden, "rf$       <pp>
1340 printc4$" "rn$" um in die Vil
la einzudringen "rf$                 <pd>
1350 printc4$b5$"A. Postbote":prin
tc4$b5$"B. Schornsteinfeger"         <bm>
1360 printc4$b5$"C. Auto Haendler"
:printc4$b5$"D. Party Servic"        <ik>
1370 getkeya$:ifa$="a"ora$="b"ora$
="c"ora$="d"then1380:else1370        <en>
1380 ifa$=mid$("cdabd",1,1)then140
0                                    <ed>
1390 scnclr:char1,1,12,"Die Bluthu
nde haben Sie getoetet.":goto1820    <ge>
1400 color4,1:color0,1:color1,6:pu
=pu+445:printaa$;l;b$;pu             <ph>
1410 printc4$c4$" In der Villa erf
ahren Sie, dass Theo"                <ff>
1420 printc4$" 5 kg Heroin ueber d
ie Grenze"                           <li>
1430 printc4$" nach Deutschland sc
hmuggeln soll."                      <co>
1440 printc4$" Sie wollen Theo auf
 der Fahrt"                          <pm>
1450 printc4$" nach Deutschland mi
t dem Auto folgen."                  <il>
1460 printc4$c4$" "rn$" Mit welche
m Wagen werden Sie fahren? "rf$      <em>
1470 printc4$b5$"A. VW Golf GTI"     <ba>
1480 printc4$b5$"B. OPEL Omega":pr
intc4$b5$"C. FORD Scorpio":printc4
$b5$"D. BMW 525i"                    <di>
1490 getkeya$:ifa$="a"ora$="b"ora$
="c"ora$="d"then1500:else1490        <af>
1500 ifa$=mid$("dabcd",1,1)then154
0                                    <dk>
1510 scnclr:char1,4,9,"Auf der Aut
obahn ist Ihnen der"                 <ea>
1520 char1,4,11,"Reifen geplatzt.
Sie haben"                           <be>
1530 char1,4,13,"keinen Reserverei
fen bei.":goto2120                   <hc>
1540 color4,6:color0,6:color1,1:pu
=pu+350:printaa$;l;b$;pu             <fm>
1550 printc4$c4$c4$" Theo faehrt a
uf ein abgesperrtes"                 <gf>
1560 printc4$" Grundstueck. Wo ein
e unbenutzte,"                       <bg>
1570 printc4$" baufaellige Firma s
teht."                               <ko>
1580 printc4$" Theo geht auf die F
abrikationshalle zu."                <ji>
1590 printc4$c4$" "rn$" Wie bekomm
en Sie heraus was Theo "rf$          <eb>
1600 printc4$" "rn$" dort treibt?
"rf$                                 <ao>
1610 printc4$b5$"A. Sie klettern a
uf ein Geruest"                      <ia>
1620 printc4$b5$"B. Sie klettern a
uf einige Faesser"                   <ed>
1630 printc4$b5$"C. Sie gehen die
Treppe hoch"                         <bj>
1640 getkeya$:ifa$="a"ora$="b"ora$
="c"then1650:else1640                <kg>
1650 ifa$=mid$("abcab",1,1)then168
0                                    <gh>
1660 scnclr:char1,3,9,"In dem bauf
aelligen Gebauede ist"               <di>
1670 char1,3,11,"Ihnen ein Holzbal
ke auf den Kopf":char1,3,13,"gefal
len.":goto1820                       <fi>
1680 color4,1:color0,1:color1,6:pu
=pu+250:printaa$;l;b$;pu             <of>
1690 printc4$c4$c4$" Theo uebergib
t die 5 kg Heroin einigen"           <ae>
1700 printc4$" Ganoven. Er erhaelt
 einen neuen"                        <bh>
1710 printc4$" Auftrag. Theo soll
einen Herrn Walter"                  <ba>
1720 printc4$" Fackelmann ermorden
.":printc4$c4$" "rn$" Was machen S
ie nun? "rf$                         <oc>
1730 printc4$b5$"A. Sie benachrich
tigen Herrn":printc4$b4$b4$"Fackel
```

```
mann"                                  <la>
1740 printc4$b5$"B. Sie beobachten
 Herrn Fackelmann"                     <ji>
1750 printc4$b5$"C. Sie beobachten
 Theo Goldini":printc4$b4$b4$"weit
er"                                    <dh>
1760 getkeya$:ifa$="a"ora$="b"ora$
="c"then1770:else1760                  <kh>
1770 ifa$=mid$("cbaac",1,1)then180
0                                      <mn>
1780 scnclr:char1,4,9,"Die Ganoven
 haben Sie in der"                     <pl>
1790 char1,4,11,"Fabrikationshalle
 entdeckt und":char1,4,13,"erschos
sen.":goto1820                         <oe>
1800 color4,6:color0,6:color1,1:pu
=pu+595                                <dk>
1810 goto1920                          <lo>
1820 forx=0to3000:nextx:ar=pu*6.13
5                                      <fj>
1830 printcl$c4$c4$c4$" Leider sin
d Sie bei Ihrem"                       <mj>
1840 printc4$" 'Einstellungstest'
getoetet worden."                      <ko>
1850 printc4$" Ihre Angehoerigen w
erden folgende"                        <db>
1860 printc4$" Betraege erhalten:"
left$(qd$,4)                           <jd>
1870 printusing" ###############.#
#";"Lebensversicherung",lv             <lm>
1880 printusing" ###############.#
#";"Arbeitslohn",ar:g=ar+lv            <an>
1890 print" ----------------------
---------------"                       <fh>
1900 printusing" ###############.#
#";"Gesamt",g:printc4$c4$c4$;e$        <oi>
1910 getkeya$:ifa$=" "then110:else
1910                                   <ah>
1920 ar=pu*8.715:s=pu*1.935:color4
,1:color0,1:color1,3                   <ho>
1930 printcl$left$(qd$,5)" *******
*********************************":p
oke3313,42                             <cc>
1940 poke3350,42:printc4$" *"b6$"H
erzlichen Glueckwunsch!"b6$"*":pok
e3393,42:poke3430,42                   <ho>
1950 printc4$" * Sie haben Ihren '
Einstellungstest' *":poke3473,42:p
oke3510,42                             <hg>
1960 printc4$" *"b5$"erfolgreich a
bgeschlossen."b5$"*":poke3553,42:p
oke3590,42                             <me>
1970 printc4$" *"b2$"Sie bekommen
einen Arbeitsplatz"b3$"*":poke3633
,42:poke3670,42                        <pc>
1980 printc4$" *"d$" bei Interpol.
 "d$" *":poke3713,42:poke3750,42       <gl>
1990 printc4$" *******************
*******************":printleft$(qd
$,4);e$                                <hb>
2000 getkeya$:ifa$=" "then2010:els
e2000                                  <dj>
2010 printcl$c4$" Da Sie Theo Gold
ini beschattet haben"                  <bp>
2020 printc4$" erhalten Sie folgen
de Betraege:":printc4$                 <go>
2030 printusingb5$+"########.##";"
Arbeitslohn",ar                        <gf>
2040 printusingb5$+"########.##";"
Spesen",s                              <bl>
2050 printusingb5$+"########.##";"
Belohnung",be                          <bb>
2060 printb5$"-------------------
-------"                               <nj>
2070 y=ar+s+be:printusingb5$+"####
####.##";"Gesamt",y                    <ho>
2080 printleft$(qd$,4)" Nun haben
Sie erst einmal 4 Wochen"              <bi>
2090 printc4$" Urlaub. Danach hole
n Sie bitte bei"                       <lg>
2100 printc4$" Interpol einen neue
n Auftrag ab."c4$c4$:printe$           <lc>
2110 getkeya$:ifa$=" "then110:else
2110                                   <im>
2120 forx=0to3000:nextx:printcl$c4
$c4$" Bedauerlicher weise muss ich
 Ihnen"                                <no>
2130 printc4$" mitteilen das Sie f
uer einen":printc4$" Arbeitsplatz
bei Interpol nicht"                    <ba>
2140 printc4$" geeignet sind. Fuer
 Ihre Muehe werden"                    <gm>
2150 printc4$" folgende Betraege a
uf Ihrem Konto":printc4$" gutgesch
rieben:"c4$c4$                         <jc>
2160 ar=pu*5.737:printusingb5$+"##
######.##";"Arbeitslohn",ar            <nh>
2170 s=pu*1.715:printusingb5$+"###
#####.##";"Spesen",s                   <pp>
2180 printb5$"-------------------
-------":g=ar+s                        <md>
2190 printusingb5$+"########.##";"
Gesamt",g:printleft$(qd$,4);e$         <gj>
2200 getkeya$:ifa$=" "then110:else
2200                                   <oi>
2210 rem nachspann ==============      <hf>
2220 rem * farbcodes/steuercodes *     <ig>
2230 c4$=chr$(017):rn$=chr$(018)       <mn>
2240 fl$=chr$(130):fo$=chr$(132)       <bf>
2250 rf$=chr$(146):cl$=chr$(147)       <eh>
2260 rem *** zeichensatz/graphik *     <cl>
2270 s2$=chr$(160):d$=chr$(32)         <ik>
2280 rem ********* zeichenfolgen *     <hl>
2290 for q=1 to 40                     <hh>
2300 qd$=qd$+c4$                       <fg>
2310 next q                            <jc>
```

INTERPOL-BESCHATTUNG — 5 —

```
2320 b2$=d$+d$:b3$=b2$+d$                <ka>
2330 b4$=b3$+d$:b5$=b4$+d$               <hn>
2340 b6$=b5$+d$:d$=b5$+b5$:return        <np>
2350 rem ========================        <cb>
2360 rem 12277 bytes memory              <me>
2370 rem 10809 bytes program             <en>
2380 rem 00161 bytes variables           <db>
2390 rem 00000 bytes arrays              <ei>
2400 rem 00495 bytes strings             <gi>
2410 rem 00812 bytes free (0)            <ni>
2420 rem ========================        <pj>
```

Merkur



Alle, die ihr Geld nicht aus dem Fenster werfen und trotzdem den Nervenkitzel eines Geldautomatenspiels erleben wollen, sind mit dem Spiel Merkur am besten bedient. Hier können Sie die Chancen eines Gewinns solcher Automaten simulieren.

Das Programm simuliert einen Geldspielautomaten. Nach der Titelmelodie und dem Aufbau des Titelbildes werden Sie gefragt, ob Sie eine Spielanleitung wünschen. Bestätigen Sie dies mit J für Ja oder N für Nein. Bei „Ja" wird die Spielanleitung ausgegeben. Nachdem diese vollständig gelesen wurde, fragt Sie das Programm, ob Sie alles verstanden haben. Wenn Sie nicht mit J für Ja antworten, beginnt die Anleitung wieder von vorne. Nachdem Sie also die Anleitung beendet haben, beginnt das Spiel. Hier müssen Sie ca. 30 Sekunden warten, da der Computer die Bildschirmdaten berechnet. Dann wird die Spielautomat-Grafik aufgebaut. Zuerst müssen Sie nun über Ihr Startkapital entscheiden. Zur Auswahl stehen entweder fünf oder zehn Mark. Wählen Sie den Betrag durch Betätigen der Taste 1 und 2 aus. Auf dem Bild des Spielautomaten sehen Sie drei Walzen. Zur Bedienung des Automaten stehen Ihnen nun folgende Möglichkeiten zur Verfügung, die dem Original gleichen:
1. Die linke Walze läßt sich pro Spiel einmal neu starten, falls Ihnen der Inhalt nicht gefällt. Hierfür muß die Taste F1 gedrückt werden, solange am unteren Bildschirmrand „STARTEN" angezeigt wird.
2. Die beiden rechten Walzen können Sie vorzeitig stoppen, wenn Sie die Taste F3 betätigen, solange das Feld „STOPPEN" aufleuchtet.

Folgende Gewinnmöglichkeiten gibt es:
1. Ist in allen drei Walzen einmal der selbe Wert enthalten, so haben Sie einen Gewinn i. der Höhe dieses Wertes. Das Programm fragt dabei ab, ob Sie auf Risiko spielen wollen. Beantworten Sie mit N für Nein, wird der Gewinn übernommen. Bei J für Ja kommen Sie in die Risikoausspielung, bei der Sie Ihren Gewinn entweder verdoppeln oder aber verlieren können. Um bei der Risikoausspielung zu stoppen, drücken Sie F3.
2. Befindet sich auf der dritten Walze ein Merkur, gewinnen Sie ebenfalls.
3. Befindet sich auf jeder Walze ein Merkur, kommen Sie in die Sonderausspielung, in der Sie bis zu 100 Sonderspiele erhalten können.
4. Bei Sonderspielen erhöht sich jeder Gewinn auf drei Mark. Erscheint in Sonderspielen neben der dritten Walze eine Linie, bringt dies ebenfalls drei Mark Gewinn.

Nach jedem Spiel können Sie sich den Betrag mit der Taste F7 „auszahlen" lassen. Den Betrag, der Ihnen noch zur Verfügung steht, sehen Sie am rechten oberen Bildschirmrand, darunter werden die eventuellen Sonderspiele angezeigt.
Haben Sie kein Geld mehr zur Verfügung, meldet der Computer, daß Sie pleite sind. Sie haben dann die Möglichkeit, nochmals zu beginnen oder das Spiel zu beenden. □

```
10 rem merkur===================p4 <ig>
20 rem (p) commodore welt team     <ho>
30 rem ========================== <ng>
40 rem (c) by b. harrer        7.0 <pi>
50 rem (v) by a. phillips      3.5 <ae>
60 rem                             <ah>
70 rem basic v3.5                  <nl>
80 rem plus4 (c16/116 + 64 kb)     <fd>
90 rem ========================== <jg>
100 scnclr:graphic1,1:gosub5890    <og>
110 key1,"w":key3,"s":key8,"x"     <fj>
120 x$=chr$(187):y$=chr$(161)      <fg>
130 z$=chr$(079)                   <db>
140 color0,1:color2,1:color4,1:col
or1,8,7                            <aj>
150 box1,1,1,318,198               <nb>
160 box1,3,3,316,196               <ed>
170 char1,9,2,"m e r k u r"        <nn>
180 char1,4,5,"ein programm von b.
harrer"                            <pg>
190 char1,9,7,"(c) 1988"           <ac>
200 rem ------------------------- <bf>
210 rem  titelmusik                <eo>
220 rem ------------------------- <de>
230 vol8:do:readx,y:sound1,x,y:loo
puntilx=0:goto290                  <fd>
240 data596,15,704,45,704,15,739,1
5,770,15,739,15,704,15,685,15,739,
45                                 <nm>
250 data596,15,596,15,739,45,739,1
5,770,15,779,15,770,15,739,15,704,
15,770,45                          <la>
260 data704,15,704,15,779,45,779,1
5,810,15,834,15,810,15,779,15,770,
45                                 <ha>
270 data596,15,596,15,739,45,739,1
5,770,15,779,15,770,15,739,15,704,
45                                 <fj>
280 data0,1                        <ph>
290 color1,8,7                     <ik>
300 char1,8,12,"anleitung ?",1     <fc>
310 getkeyan$                      <hm>
320 ifan$="j"then5610              <ib>
330 ifan$<>"n"then300              <kp>
340 scnclr:color1,8,7              <jb>
350 char1,5,20,"bitte ca. 25 sek.
geduld !"                          <kl>
360 char1,3,22,"shapes werden defi
niert!",1                          <fj>
370 scnclr                         <mo>
380 graphic 1,1                    <be>
390 rem ------------------------- <fj>
400 rem  shapes fuer titel         <in>
410 rem ------------------------- <oa>
420 rem ** m **                    <po>
430 draw,3,19to3,1to10,12to16,1to1
6,19                               <ie>
440 sshapem$,0,0,23,20             <bf>
450 scnclr                         <gp>
460 rem ** e **                    <me>
470 draw,3,1to3,19to13,19          <ee>
480 draw,3,10to9,10:draw1,3,1to13,
1                                  <ca>
490 sshapee$,0,0,23,20             <ch>
500 scnclr                         <nd>
510 rem ** r **                    <in>
520 draw1,5,1to5,19                <ol>
530 circle1,5,7,8,6,0,180          <fh>
540 draw1,7,12to14,19              <fd>
550 sshaper$,0,0,23,20             <bh>
560 scnclr                         <em>
570 rem ** k **                    <ia>
580 draw1,4,1to4,19:draw1,4,12to12
,1                                 <ik>
590 draw1,4,8to12,19               <kg>
600 sshapek$,0,0,23,20             <ab>
610 scnclr                         <la>
620 rem ** u **                    <ce>
630 circle1,14,14,5,5,90,270       <fb>
640 draw1,9,14to9,1:draw1,19,14to1
9,1                                <kb>
650 sshapeu$,0,0,23,20             <ca>
660 scnclr                         <bf>
670 rem ** diamant **              <jm>
680 graphic 1,1                    <oh>
690 draw1,5,1to19,1to21,5to3,5to5,
1to9,5to12,1to15,5to19,1           <bc>
700 draw1,3,5to12,20to21,5         <go>
710 draw1,7,5to12,20to18,5         <ek>
720 draw1,12,5to12,20              <ke>
730 sshapedi$,0,0,23,20            <pi>
740 scnclr                         <lf>
750 rem ------------------------- <ch>
760 rem  geld-und spielesshapes    <ld>
770 rem ------------------------- <ob>
780 graphic 1,1                    <dc>
790 rem ** 1 **                    <gp>
800 draw1,12,16to12,2to8,5         <kh>
810 sshapez$(1),0,0,23,20:scnclr   <lo>
820 rem ** 2 **                    <og>
830 circle1,13,5,3,3,270,123       <aa>
840 draw1,16,7to8,15to17,15        <ej>
850 sshapez$(2),0,0,23,20:scnclr   <pn>
860 rem ** 3 **                    <oi>
870 circle1,12,5,3,3,320,180       <md>
880 circle1,12,12,4,4,340,240      <nl>
890 sshapez$(3),0,0,23,20:scnclr   <eb>
900 rem ** 4 **                    <bc>
910 draw1,6,2to6,9to16,9           <ma>
920 draw1,12,5to12,16              <fc>
930 sshapez$(4),0,0,23,20:scnclr   <pp>
940 rem ** 5 **                    <be>
950 circle1,12,12,4,4,310,220      <ej>
960 draw1,9,9to9,2to16,2           <pi>
970 sshapez$(5),0,0,23,20:scnclr   <lm>
980 rem ** 6 **                    <ao>
```

```
990 circle1,12,9,6,7,190,25            <gm>
1000 circle1,11,12,4,4,300,210         <ac>
1010 sshapez$(6),0,0,23,20:scnclr      <ak>
1020 rem ** 7 **                       <ca>
1030 draw1,6,3to15,3to7,17             <oa>
1040 draw1,8,10to15,10                 <ma>
1050 sshapez$(7),0,1,23,20:scnclr      <dm>
1060 rem ** 8 **                       <el>
1070 circle1,12,6,3,3                  <gn>
1080 circle1,12,14,4,4                 <cl>
1090 sshapez$(8),0,1,23,20:scnclr      <ae>
1100 rem ** 9 **                       <dn>
1110 circle1,12,7,4,4                  <eb>
1120 circle1,11,10,6,7,0,195           <ma>
1130 sshapez$(9),0,1,23,20:scnclr      <ln>
1140 rem ** 0 **                       <ao>
1150 circle1,12,9,5,7                  <cl>
1160 sshapez$(0),0,0,23,20:scnclr      <cl>
1170 rem -------------------------     <dg>
1180 rem  walzenshapes                 <pf>
1190 rem -------------------------     <am>
1200 rem ** 40 **                      <oc>
1210 draw1,3,2to3,8to10,8              <id>
1220 draw1,7,4to7,15                   <fn>
1230 circle1,16,9,3,6                  <ki>
1240 sshapew1$,0,0,23,20:scnclr        <ba>
1250 rem ** 60 **                      <oh>
1260 circle1,6,15,3,3                  <lg>
1270 circle1,6,10,3,6,240,25           <mm>
1280 circle1,16,11,3,7                 <lo>
1290 sshapew2$,0,2,23,22:scnclr        <hn>
1300 rem ** 80 **                      <fj>
1310 circle1,6,5,3,3:circle1,6,13,
4,4                                    <ap>
1320 circle1,17,9,3,7                  <di>
1330 sshapew3$,0,0,23,20:scnclr        <gd>
1340 rem ** 1.20 **                    <ig>
1350 char1,0,1,"120"                   <dh>
1360 circle1,3,19,2,1:circle1,11,1
9,2,1                                  <eb>
1370 circle1,19,19,2,1:draw1,1,4to
21,4                                   <im>
1380 sshapew4$,0,2,23,20:scnclr        <ni>
1390 rem ** 3.- **                     <cb>
1400 circle1,6,6,3,3,325,190           <kn>
1410 circle1,6,12,4,4,0,210            <en>
1420 circle1,18,6,1,1                  <in>
1430 box1,18,9,21,11                   <bo>
1440 sshapew5$,0,0,23,20:scnclr        <ng>
1450 rem ** <*> **                     <jd>
1460 graphic1,1                        <ni>
1470 draw1,1,1to6,3to6,1to8,3to9,1
to10,3to11,1to12,3                     <gk>
1480 draw1,12,3to14,1to14,3to16,1t
o16,3to21,1to18,5to23,7                <fh>
1490 draw1,23,7to18,8to23,10to18,1
1to23,16to18,14to18,17                 <fg>
1500 draw1,18,17to16,14to14,17to14
,14to12,17to11,15to9,17to8,15          <ne>
1510 draw1,8,15to1,18to7,13to1,12t
o7,10to0,8to6,6to1,1                   <oj>
1520 circle1,9,7,1,1:circle1,15,7,
1,1                                    <nd>
1530 draw1,9,11to10,12to14,12to15,
11                                     <jb>
1540 draw1,12,9                        <cc>
1550 sshapew6$,0,0,23,20:scnclr        <gj>
1560 rem -------------------------     <im>
1570 rem  geldstueck                   <hp>
1580 rem -------------------------     <an>
1590 graphic 1,1                       <aa>
1600 circle1,10,9,9,9:circle1,10,1
1,3,3,320,230                          <fm>
1610 draw1,8,9to8,4to13,4              <io>
1620 circle1,13,9,9,9,340,200          <ln>
1630 sshapem5$,0,0,23,20:scnclr        <jd>
1640 sshapen$,0,0,23,20                <cm>
1650 rem -------------------------     <oi>
1660 rem  bildaufbau                   <ng>
1670 rem -------------------------     <jf>
1680 box1,4,4,177,198                  <ba>
1690 box1,1,40,180,198                 <gl>
1700 box1,1,1,180,198:color1,16,7      <fg>
1710 box,12,43,41,111                  <gc>
1720 box,45,53,74,102                  <bb>
1730 box,78,64,107,91:color1,8,7       <mn>
1740 draw1,1,160to180,160              <id>
1750 draw1,140,1to140,160              <ea>
1760 box1,172,10,174,25                <pa>
1770 box1,162,11,170,24                <em>
1780 fora=1to6                         <kc>
1790 draw1,162,11+2*ato170,11+2*a:
nexta                                  <od>
1800 circle1,151,14,6,6                <jp>
1810 circle1,151,15,2,2,320,240        <kc>
1820 circle1,151,20,5,5,90,270         <cp>
1830 circle1,151,19,2,3,90,270         <dj>
1840 draw1,149,14to149,11to153,11      <ha>
1850 circle1,151,25,4,4,90,270         <dn>
1860 circle1,151,24,2,3,90,270         <je>
1870 draw1,140,40to180,40              <om>
1880 circle1,151,55,2,2,180,0          <of>
1890 circle1,168,55,2,2,0,180          <pk>
1900 draw1,151,53to168,53              <nc>
1910 draw1,151,57to168,57              <ak>
1920 color1,9,7:fora=1to24             <ce>
1930 draw1,145,60+4*ato175,60+4*a:
nexta                                  <pg>
1940 color1,12,5                       <ea>
1950 circle1,40,179,35,10              <fm>
1960 paint1,35,170                     <fo>
1970 circle1,122,179,35,10:color1,
14,7                                   <jl>
1980 paint1,120,170                    <mn>
1990 circle1,170,170,4,4               <ob>
2000 circle1,170,170,1,2,130,50:co
```

```
lor1,14,7                                   <fd>
2010 gshapem$,10,10,1:gshapee$,32,
10,1                                        <en>
2020 gshaper$,49,10,1:gshapek$,69,
10,1                                        <hn>
2030 gshapeu$,83,10,1:gshaper$,106
,10,1                                       <nn>
2040 color1,13,7:gshapedi$,15,130,
1:gshapei$,75,110,1                         <fc>
2050 color1,5,7:gshapedi$,45,120,1
:gshapedi$,105,100,1:color1,8,7             <fk>
2060 rem ------------------------           <gn>
2070 rem  startkapital                      <hn>
2080 rem ------------------------           <no>
2090 char1,25,3,"startgeld:",1              <bm>
2100 char1,25,6,"1 ="+b2$+"5 dm"            <pb>
2110 char1,25,8,"2 = 10 dm"                 <gf>
2120 char1,25,10,"bitte":char1,25,
11,"waehlen !"                              <mi>
2130 getkeyst$:char1,25,3,b$                <ii>
2140 char1,25,6,b5$+b4$                     <hf>
2150 char1,25,8,b5$+b4$                     <ho>
2160 char1,25,10,b5$:char1,25,11,b
5$+b4$                                      <hg>
2170 ifj=1thenj=0:goto2090                  <ii>
2180 p=0:m=0:m1=0:m2=0:s=0:s1=0:s2
=0                                          <nk>
2190 ifst$="1"then2220                      <pl>
2200 ifst$="2"then2340                      <mb>
2210 goto2090                               <no>
2220 rem ------------------------           <fb>
2230 rem  5 dm                              <ak>
2240 rem ------------------------           <he>
2250 m=5:p=0                                <oe>
2260 gshapem5$,240,40:do:readx,y:s
ound1,x,y                                   <lb>
2270 loop until x=0                         <gm>
2280 data453,30,453,30,571,30,571,
30                                          <ka>
2290 data516,15,453,15,516,15,571,
15                                          <ah>
2300 data453,30,262,30                      <dk>
2310 data0,1                                <ff>
2320 gshapen$,240,40                        <gb>
2330 zeit$="000001":ti$="000000":d
ountilti$=zeit$:loop:goto2510               <dj>
2340 rem ------------------------           <dh>
2350 rem  10 dm                             <ph>
2360 rem ------------------------           <eb>
2370 m1=1:m=0:p=0                           <fe>
2380 gshapem5$,240,40:gshapem5$,28
0,40                                        <mn>
2390 do:reada,b:sound1,a,b:loopunt
ila=0                                       <ek>
2400 data643,30,643,30,685,30,643,
30                                          <ap>
2410 data798,45,798,15,798,45,798,
45,798,15                                   <ne>
2420 data770,30,739,30,770,45,685,
45                                          <pd>
2430 data0,1                                <no>
2440 gshapen$,240,40:gshapen$,280,
40                                          <hk>
2450 do:reada,b:sound1,a,b:loopunt
ila=0                                       <gn>
2460 data643,30,643,30,685,30,643,
30                                          <ic>
2470 data798,45,798,15,798,45,798,
45,798,15                                   <ig>
2480 data770,30,739,30,770,45,685,
45                                          <lo>
2490 data0,1                                <jk>
2500 zeit$="000001":ti$="000000":d
ountilti$=zeit$:loop                        <pa>
2510 rem ------------------------           <kj>
2520 rem  start                             <bh>
2530 rem ------------------------           <dl>
2540 printchr$(14):getau$                   <bi>
2550 ifau$="x"then5480                      <gp>
2560 char1,14,8," ":char1,14,9," "
:char1,14,10," ":color1,8,7                 <ag>
2570 gshapen$,15,45,1:gshapen$,15,
66,1:gshapen$,15,87,1                       <ae>
2580 gshapen$,15,87,1:gshapen$,48,
55,1                                        <nc>
2590 gshapen$,48,76,1:gshapen$,81,
66,1                                        <ki>
2600 rem ------------------------           <an>
2610 rem  berechnungen                      <cj>
2620 rem ------------------------           <kb>
2630 ifm<1andp<30andm1=0andm2=0the
n5360                                       <fb>
2640 x=0:x1=0:x2=0                          <kd>
2650 p=p-30                                 <on>
2660 ifs=0ands1=0ands2=0then2750            <nb>
2670 ifs>0ors1>0ors2>0thens=s-1             <hn>
2680 ifs<0thens=s+10:s1=s1-1                <gp>
2690 ifs>10thens=s-10:s1=s1+1               <em>
2700 ifs=10thens=0:s1=s1+1                  <of>
2710 ifs1<0thens1=s1+10:s2=s2-1             <hj>
2720 ifs1>10thens1=s1-10:s2=s2+1            <lc>
2730 ifs1=10thens1=0:s2=s2+1                <hn>
2740 ifs2=>1ands1=>2thens2=1:s1=2:
s=0                                         <pc>
2750 ifp<0thenp=p+100:m=m-1                 <no>
2760 ifp>100thenp=p-100:m=m+1               <ab>
2770 ifp=100thenp=0:m=m+1                   <ih>
2780 ifm<0thenm=m+10:m1=m1-1                <dl>
2790 ifm>10thenm=m-10:m1=m1+1               <md>
2800 ifm=10thenm=0:m1=m1+1                  <cp>
2810 ifm1<0thenm1=m1+10:m2=m2-1             <ki>
2820 ifm1>10thenm1=m1-10:m2=m2+1            <eo>
2830 ifm1=10thenm1=0:m2=m2+1                <hd>
2840 p=int(p*10↑0+.5)/10^0                  <od>
2850 color1,5,7                             <aa>
2860 char1,25,2,"b e t r a g :",1           <hp>
```

```
2870 color1,3,7                          <on>
2880 char1,25,10,"s p i e l e :",1       <ne>
2890 color1,15,7                         <dj>
2900 char1,25,17,"r i s i k o :",1       <dg>
2910 color1,8,7                          <bh>
2920 rem -------------------------       <gl>
2930 rem  geldanzeige                    <cp>
2940 rem -------------------------       <bg>
2950 gshapez$(0),294,38,1                <dm>
2960 gshapez$(p/10),272,38,1             <do>
2970 gshapez$(m),234,38,1                <ea>
2980 gshapez$(m1),212,38,1               <ie>
2990 gshapez$(m2),190,38,1:ifs>0or
s1>0ors2>0thencolor1,13,7                <do>
3000 gshapez$(s),258,102,1               <co>
3010 gshapez$(s1),235,102,1              <ce>
3020 gshapez$(s2),214,102,1:color1
,2,7                                     <hc>
3030 char1,33,6,z$                       <bo>
3040 rem -------------------------       <dl>
3050 rem  auswahl 1. walze               <jn>
3060 rem -------------------------       <ek>
3070 a=int(rnd(0)*5)+1                   <dd>
3080 onagosub3090,3110,3130,3150,3
170:goto3190                             <pa>
3090 color1,2,7:gshapew1$,15,46,1:
color1,15,5                              <oa>
3100 gshapew2$,15,66,1:color1,9,5:
gshapew3$,15,87,1:return                 <kp>
3110 color1,15,5:gshapew2$,15,46,1
:color1,9,5                              <ob>
3120 gshapew3$,15,66,1:color1,6,5:
gshapew4$,15,87,1:return                 <on>
3130 color1,12,7:gshapew6$,15,46,1
:color1,16,5                             <md>
3140 gshapew5$,15,66,1:color1,15,5
:gshapew2$,15,87,1:return                <jp>
3150 color1,12,7:gshapew6$,15,46,1
:color1,2,7                              <kl>
3160 gshapew1$,15,66,1:color1,12,7
:gshapew6$,15,87,1:return                <fa>
3170 color1,12,7:gshapew6$,15,46,1
:gshapew6$,15,66                         <io>
3180 gshapew6$,15,87,1:return            <nk>
3190 ifwe=1thenwe=0:goto3310             <hf>
3200 color1,8,7:char1,2,22,"*start
*":getp$                                 <cf>
3210 sound1,685,5:sound1,596,5:sou
nd1,739,5:goto3220                       <hl>
3220 zeit$="000001"                      <ea>
3230 ti$="000000"                        <ac>
3240 do until ti$=zeit$                  <dl>
3250 loop:we=1                           <cc>
3260 getp$:ifp$="w"thenchar1,2,22,
b4$+b3$                                  <eh>
3270 ifp$="w"thengshapen$,15,46,1:
gshapen$,15,66,1:gshapen$,15,87,1        <nd>
3280 ifp$="w"thensound1,685,5:soun
d1,596,5:sound1,739,5:goto3040           <nl>
3290 ifp$<>"w"thenwe=0                   <gj>
3300 rem -------------------------       <hn>
3310 rem  auswahl 2. walze               <dd>
3320 rem -------------------------       <oc>
3330 color1,2,7:char1,2,22,b4$+b3$
:char1,13,22,"*stop*"                    <am>
3340 geto$:zeit$="000001":ti$="000
000":dountilti$=zeit$:loop               <ap>
3350 geto$:ifo$="s"then3370              <nm>
3360 ifo$<>"s"thensound1,596,5           <pf>
3370 b=int(rnd(0)*5)+1                   <mk>
3380 char1,13,22,b6$:sound1,383,5:
sound1,596,5:sound1,516,5                <dk>
3390 onbgosub3400,3410,3420,3430,3
440:goto3450                             <gl>
3400 color1,2,7:gshapew1$,48,56,1:
color1,15,5:gshapew2$,48,78,1:retu
rn                                       <ba>
3410 color1,9,5:gshapew3$,48,56,1:
color1,6,5:gshapew4$,48,78,1:retur
n                                        <pl>
3420 color1,16,5:gshapew5$,48,56,1
:color1,12,7:gshapew6$,48,78,1:ret
urn                                      <cl>
3430 color1,2,7:gshapew1$,48,56,1:
color1,16,5:gshapew5$,48,78,1:retu
rn                                       <ff>
3440 color1,12,7:gshapew6$,48,56,1
:gshapew6$,48,78,1:return                <fl>
3450 rem -------------------------       <bg>
3460 rem  auswahl 3. walze               <kn>
3470 rem -------------------------       <nj>
3480 color1,2,7:char1,13,22,"*stop
*"                                       <ii>
3490 getu$:zeit$="000001":ti$="000
000":dountilti$=zeit$:loop               <gn>
3500 getu$:ifu$="s"then3520              <ko>
3510 ifu$<>"s"thensound1,739,5           <ec>
3520 c=int(rnd(0)*6)+1                   <de>
3530 char1,13,22,b6$:sound1,739,5:
sound1,798,5:sound1,643,5                <pp>
3540 oncgosub3550,3560,3570,3580,3
590,3600:goto3610                        <md>
3550 color1,2,7:gshapew1$,81,68,1:
return                                   <ad>
3560 color1,15,5:gshapew2$,81,68,1
:return                                  <nn>
3570 color1,9,5:gshapew3$,81,68,1:
return                                   <hg>
3580 color1,6,5:gshapew4$,81,68,1:
return                                   <jk>
3590 color1,16,5:gshapew5$,81,68,1
:return                                  <pp>
3600 color1,12,7:gshapew6$,81,68,1
:return                                  <ki>
3610 rem -------------------------       <no>
3620 rem  sonderspielegewinn             <fa>
```

```
3630 rem -------------------------- <fh>
3640 k=int(rnd(0)*4)                <hj>
3650 ifk=0ork=1thenchar1,14,8,x$:c
har1,14,9,y$:char1,14,10,y$         <ap>
3660 color1,8,7                     <fe>
3670 rem -------------------------- <gb>
3680 rem  gewinnauswertung          <lj>
3690 rem -------------------------- <oi>
3700 ifa=1andb=1andc=1ora=4andb=1a
ndc=1theng$="0.40":dm=0:pf=40:goto
3930                                <if>
3710 ifnot(a=4andb=1andc=6ora=1and
b=4andc=1ora=1andb=4andc=6)then373
0                                   <il>
3720 g$="0.40":dm=0:pf=40:goto3930 <ob>
3730 ifa=4andb=4andc=6ora=4andb=4a
ndc=1theng$="0.40":dm=0:pf=40:goto
3930                                <lo>
3740 ifnot(a=1andb=1andc=2ora=1and
b=1andc=6ora=3andb=1andc=2)then376
0                                   <fc>
3750 g$="0.60":dm=0:pf=60:goto3930 <fk>
3760 ifnot(a=3andb=1andc=6ora=2and
b=1andc=2ora=2andb=1andc=6)then378
0                                   <ci>
3770 g$="0.60":dm=0:pf=60:goto3930 <ha>
3780 ifnot(a=1andb=2andc=3ora=1and
b=2andc=6ora=2andb=2andc=3)then380
0                                   <fp>
3790 g$="0.80":dm=0:pf=80:goto3930 <gc>
3800 ifa=2andb=2andc=4ora=2andb=2a
ndc=6theng$="1.20":dm=1:pf=20:goto
3930                                <gb>
3810 ifa=3andb=3andc=5ora=3andb=4a
ndc=5ora=3andb=4andc=6theng$="3.00
":dm=3:pf=0:goto3930                <je>
3820 rem -------------------------- <le>
3830 rem  sonderspiele              <am>
3840 rem -------------------------- <ii>
3850 ifa=3andb=3andc=6ora=3andb=5a
ndc=6ora=4andb=3andc=6thenifs2<1th
en4380                              <cp>
3860 ifa=4andb=5andc=6ora=5andb=3a
ndc=6thenifs2<1then4530             <ki>
3870 ifa=5andb=5andc=6thenifs2<1an
ds1<7then4680                       <ga>
3880 ifc=6theng$="0.30":dm=0:pf=30
:goto4680                           <ed>
3890 ifk=0ork=1then3910             <ld>
3900 goto3920                       <aa>
3910 ifs>0ors1>0ors2>0theng$="3.00
":dm=3:pf=0:goto3930                <nf>
3920 dm=0:pf=0:sound1,475,100:goto
2510                                <an>
3930 rem -------------------------- <jm>
3940 rem  gewinnanzeige             <pa>
3950 rem -------------------------- <ea>
3960 ifs>0ors1>0ors2>0theng$="3.00
":dm=3:pf=0                         <di>
3970 char1,27,20,"gewinn:"          <ne>
3980 char1,28,22,g$:sound1,596,10:
sound1,516,10                       <ef>
3990 char1,27,24,"risiko ?",1       <bg>
4000 getr$:ifr$=""then4000          <fc>
4010 ifr$="n"thenm=m+dm:p=p+pf:got
o4040                               <jh>
4020 ifr$="j"then4120               <bp>
4030 goto3990                       <fp>
4040 sound1,704,10                  <pd>
4050 char1,27,20," "+b$             <gl>
4060 char1,27,22," "+b$             <hk>
4070 char1,26,24," "+b$             <hn>
4080 goto2510                       <ib>
4090 rem -------------------------- <ci>
4100 rem  riskieren                 <gb>
4110 rem -------------------------- <im>
4120 char1,28,22,b6$:char1,27,20,b
4$+b4$                              <ea>
4130 char1,27,24," "+b$             <eo>
4140 char1,27,20,"0x":sound1,600,1
0:char1,27,20,b2$                   <ma>
4150 char1,32,20,"2x":sound1,250,1
0:char1,32,20,b2$                   <cd>
4160 getf$:iff$="s"then4180         <hc>
4170 goto4140                       <nc>
4180 d=int(rnd(0)*4)                <jo>
4190 ifd=0ord=1thenchar1,27,20,"0x
",1:goto4210                        <gf>
4200 ifd=2ord=3thengoto4230         <ab>
4210 char1,27,22,"leider":char1,27
,24,"nichts !":sound1,897,50:sound
1,864,50                            <kj>
4220 char1,27,22,b4$+b3$:char1,27,
24,b4$+b4$:char1,27,20,b2$:goto251
0                                   <ae>
4230 rem -------------------------- <dg>
4240 rem  2 x                       <pp>
4250 rem -------------------------- <am>
4260 char1,32,20,"2x",1             <dg>
4270 ifg$="0.30"theng1$="0.60":dm=
0:pf=60                             <nc>
4280 ifg$="0.40"theng1$="0.80":dm=
0:pf=80                             <mp>
4290 ifg$="0.60"theng1$="1.20":dm=
0:pf=20                             <gk>
4300 ifg$="0.80"theng1$="1.60":dm=
1:pf=60                             <fe>
4310 ifg$="1.20"theng1$="2.40":dm=
2:pf=40                             <hp>
4320 ifg$="3.00"theng1$="5 spiele"
:dm=0:pf=0:s=s+5                    <cn>
4330 char1,27,22,"gewinn:",1        <jf>
4340 char1,27,24,g1$:sound1,596,50
:sound1,739,50                      <le>
4350 char1,27,22,b$                 <ok>
4360 char1,27,24,b5$+b4$:char1,32,
```

```
20,b2$                                  <bm>
4370 m=m+dm:p=p+pf:goto2510             <ij>
4380 rem -------------------------      <dh>
4390 rem  3 + 4 sterne                  <md>
4400 rem -------------------------      <eb>
4410 color1,14,7                        <bb>
4420 char1,25,17,"ausspielung :",1      <bh>
4430 ka=int(rnd(0)*5):color1,8,7        <cf>
4440 char1,28,19,"3":sound1,596,10
:char1,28,19," "                        <eo>
4450 char1,32,19,"12":char1,32,19,
b2$:sound1,704,20                       <cl>
4460 char1,30,19,"6":char1,30,19,"
"                                       <kb>
4470 getr$:ifr$="s"then4490             <bf>
4480 goto4440                           <nf>
4490 onkagoto4500,4500,4510,4510,4
520                                     <ca>
4500 char1,28,19,"3",1:x=3:g$="3 s
piele":goto4860                         <jg>
4510 char1,30,19,"6",1:x=6:g$="6 s
piele":goto4860                         <kl>
4520 char1,32,19,"12",1:x1=1:x=2:g
$="12 spiele":goto4860                  <ce>
4530 rem -------------------------      <nf>
4540 rem  5 sterne                      <ff>
4550 rem -------------------------      <kj>
4560 color1,14,7                        <hl>
4570 char1,25,17,"ausspielung :",1      <fn>
4580 ma=int(rnd(0)*6):color1,8,7        <gi>
4590 char1,25,19,"6":sound1,704,10
:char1,25,19,b2$                        <mi>
4600 char1,27,19,"12":char1,27,19,
b2$                                     <dd>
4610 char1,30,19,"25":sound1,770,2
0:char1,30,19,b2$                       <ik>
4620 getv$:ifv$="s"then4640             <ll>
4630 goto4590                           <lc>
4640 onmagoto4650,4650,4650,4660,4
660,4670                                <eh>
4650 char1,25,19,"6",1:x=6:g$="6 s
piele":goto4860                         <ck>
4660 char1,27,19,"12",1:x1=1:x=2:g
$="12 spiele":goto4860                  <ma>
4670 char1,30,19,"25",1:x1=2:x=5:g
$="25 spiele":goto4860                  <bd>
4680 rem -------------------------      <kh>
4690 rem  6 sterne                      <bj>
4700 rem -------------------------      <aa>
4710 color1,15,7                        <me>
4720 char1,25,17,"ausspielung :",1      <in>
4730 ga=int(rnd(0)*3):color1,8,7        <ee>
4740 char1,25,19,"50":sound1,685,5
:char1,25,19,b2$                        <mn>
4750 char1,29,19,"100":sound1,739,
5:char1,29,19,b3$                       <ib>
4760 getkl$:ifkl$="s"then4780           <ij>
4770 goto4740                           <dm>
4780 ongagoto4790,4790,4800             <cd>
4790 char1,25,19,"50",1:x1=5:x=0:g
$="50 spiele":goto4860                  <ok>
4800 char1,29,19,"100",1                <ca>
4810 sound1,770,10:sound1,810,10        <kb>
4820 s2=s2+1                            <fb>
4830 char1,29,19,b3$:char1,25,23,b
5$+b4$                                  <nm>
4840 color1,7,7:char1,25,17,"r i s
 i k o :",1                             <bj>
4850 m=m+1:goto2510                     <gh>
4860 rem -------------------------      <md>
4870 rem  risiko spiele                 <mf>
4880 rem -------------------------      <hi>
4890 m=m+1:color1,7,7                   <ce>
4900 char1,25,17,"r i s i k o :",1
:color1,8,7                             <kh>
4910 char1,25,21,"gewinn:",1            <ne>
4920 char1,25,23,g$                     <da>
4930 char1,25,24,"risiko ?"             <li>
4940 getq$:ifq$=""then4940              <mn>
4950 ifq$="j"then5020                   <dh>
4960 ifq$<>"n"then4900                  <ki>
4970 sound1,516,10:sound1,345,10        <la>
4980 char1,25,21,b$                     <dm>
4990 char1,25,23,b$:char1,25,19,b$
+b2$                                    <fn>
5000 char1,25,24,b$+b2$                 <af>
5010 s=s+x:s1=s1+x1:goto2510            <kg>
5020 rem -------------------------      <pd>
5030 rem  riskieren                     <nf>
5040 rem -------------------------      <nf>
5050 char1,25,21," "+b$:char1,25,2
3,b$+b2$                                <jl>
5060 char1,25,24,b5$+b4$                <cb>
5070 char1,25,19,b$+b3$                 <ma>
5080 char1,27,20,"0x":sound1,897,1
0:char1,27,20,b2$                       <cj>
5090 char1,32,20,"2x":sound1,917,1
0:char1,32,20,b2$                       <ph>
5100 gete$:ife$="s"then5120             <mi>
5110 goto5080                           <ma>
5120 z=int(rnd(0)*4)+1                  <fg>
5130 onzgoto5140,5140,5200,5200         <lp>
5140 char1,27,20,"0x",1                 <ke>
5150 char1,27,22,"leider":char1,27
,24,"nichts !":sound1,810,50:sound
1,770,50                                <al>
5160 :                                  <lc>
5170 char1,27,22,b4$+b3$:char1,27,
24,b4$+b4$                              <nl>
5180 char1,27,20,b2$:goto2510           <an>
5190 rem -------------------------      <kh>
5200 rem  2 x                           <bj>
5210 rem -------------------------      <aa>
5220 char1,32,20,"2x",1                 <gi>
5230 ifg$="3 spiele"theng1$="6 spi
ele"                                    <an>
```

```
5240 ifg$="6 spiele"theng1$="12 sp
iele"                                  <ij>
5250 ifg$="12 spiele"theng1$="24 s
piele"                                 <jf>
5260 ifg$="25 spiele"theng1$="50 s
piele"                                 <cm>
5270 ifg$="50 spiele"theng1$="100
spiele":x2=1:x1=0:x=0:goto5290         <fc>
5280 x=x+x:x1=x1+x1                    <an>
5290 s=s+x:s1=s1+x1:s2=s2+x2           <gp>
5300 char1,25,24,g1$                   <og>
5310 sound1,169,15:sound1,345,15:s
ound1,453,15:sound1,596,15             <ok>
5320 char1,25,22,b5$+b4$               <mm>
5330 char1,25,24,b$+b2$                <nd>
5340 char1,32,20,b3$                   <ho>
5350 goto2510                          <ni>
5360 rem ------------------------      <oc>
5370 rem  pleite                       <pc>
5380 rem ------------------------      <la>
5390 color1,8,7                        <ik>
5400 char1,24,19,"sie sind"            <bp>
5410 char1,24,20,"pleite !"            <pg>
5420 char1,24,22,"nochmal ???",1       <eg>
5430 getkeyno$                         <dd>
5440 fora=1to24:char1,24,a,b$+b5$:
nexta                                  <ee>
5450 ifno$="n"thengraphic0:scnclr:
print"auf wiedersehen !":end           <jm>
5460 ifno$="j"then2060                 <dg>
5470 goto5390                          <ah>
5480 rem ------------------------      <ea>
5490 rem  auszahlen                    <ae>
5500 rem ------------------------      <jc>
5510 vol 8                             <fm>
5520 fora=1to24                        <cg>
5530 char1,24,a,b$+b5$:nexta           <nd>
5540 forb=1to180step24                 <od>
5550 gshapem5$,240,b:sound1,640,10
:nextb                                 <nm>
5560 forc=1to180step24                 <na>
5570 gshapen$,240,c:sound1,745,10:
nextc:                                 <co>
5580 scnclr:graphic0                   <dc>
5590 print:printtab(4)"bis zum nae
chsten mal !"                          <ec>
5600 end                               <gl>
5610 rem ------------------------      <ek>
5620 rem  anleitung                    <lf>
5630 rem ------------------------      <jp>
5640 graphic0:scnclr                   <om>
5650 char1,9,0,"a n l e i t u n g
:",1                                   <di>
5660 color1,8,7                        <eo>
5670 print:print"das programm merk
ur simuliert ein geld- spielgeraet
,":                                    <ck>
5680 print"wie es in fast jedem lo
kal an der wand"b2$"haengt."           <ci>
5690 print:print"nach der wahl des
 startkapitals beginnt das spiel."
:print"die erste ";                    <ma>
5700 print"walze kann bei nichtgef
allen"b2$"mit >f1< nachgestartet,
die beiden"                            <em>
5710 print"anderen walzen koennen
mit >f3< vor -"b3$"zeitig gestoppt
 werden."                              <aa>
5720 print:print"ein gewinn erfolg
t bei drei gleichen":                  <bi>
5730 print"betraegen oder bei merk
ur auf der"b4$b3$"dritten walze."      <hg>
5740 print:print"befindet sich auf
 jeder walze mind. ein merkur,":       <lh>
5750 print"so erfolgt eine ausspie
lung die bis zu"b2$"100 sonderspie
len erbringen kann."                   <bc>
5760 char1,8,22,">taste<",1            <ae>
5770 getkeya$                          <kb>
5780 scnclr                            <dn>
5790 print:print"in sonderspielen
wird jeder gewinn auf"b2$"3.- erho
eht.":print                            <co>

5800 print"eine linie neben der dr
itten walze"b6$"bringt in sondersp
ielen ebenso 3.- ."                    <de>
5810 print:printchr$(18)"jeder gew
inn unter 100 sonderspielen ist mi
t":                                    <dh>

5820 print:print" 'j' und 'f3' ris
kierbar!"chr$(146)                     <ho>
5830 print:print"nach halt der dri
tten walze kann man mit>f8<"b2$"au
szahlen lassen!"                       <fm>
5840 char1,8,22,"alles klar?"          <fm>
5850 getkeyd$                          <kg>
5860 ifd$="j"thengraphic1,1:goto34
0                                      <bm>

5870 scnclr:print"also nochmal !"      <dk>
5880 goto5610                          <gp>
5890 b$=chr$(32):b2$=b$+b$             <bl>
5900 b3$=b2$+b$:b4$=b3$+b$             <oo>
5910 b5$=b4$+b$:b6$=b5$+b$             <gh>
5920 b$=b5$+b5$:return                 <bh>

5930 rem ========================      <il>
5940 rem 60671 bytes memory            <op>
5950 rem 16210 bytes program           <nd>
5960 rem 00336 bytes variables         <fh>
5970 rem 00040 bytes arrays            <di>
5980 rem 01898 bytes strings           <nj>
5990 rem 12288 bytes graphic           <jg>
6000 rem 29899 bytes free (0)          <lb>
6010 rem ========================      <kc>
```

C16/Plus4: Suche Hardcopy f. Seikosha SP-180-VC. Verk. außerdem Datasette f. C64 m. Adapter f. C16/Plus4. Angebote an: Holger Borker, Hermann-Heukingstr. 7, 4450 Lingen 1

TAUSCHE U. VERKAUFE brandneue Software f. C16/Plus4 auf Tape. Ihr habt Sie bestimmt noch nicht!! (Kaufe auch.) Michael Szekely, Eberstr. 18, Dortmund/Eving

VERKAUFE: Multiprommer V 1.5, neu, VB 120,– DM; Pulsar-Eprommer, def., VB 50,– DM; Final Cart. VB 40,– DM; Simons Basic-Modul VB

STAR-GEMINI 10x: Drucker, neuwertig m. Garantie, Centronics Par.-Anschluß u. Commodore-Interface DM 640,–, für nur DM 440,– zu verk. Tel. 06201/69189

Ich verkaufe 60 Spiele für nur 30,– DM od. 100 Spiele f. 50,– DM. Schickt das Geld an: Alexander Käss, Hummelstr. 7, 7987 Weingarten (nur Disk.).

FÜR PLUS4: Tausche Anwend.-Prog. (bes. Schule) auf Disk. od. Kass. Liste an: Johannes Maass, Am Glaskopf 59, 5244 Daaden 2

Verkaufe Computerhefte ab 1986! Verkaufe auch meine PDS-Software f. C16! 6 Disk. od. Kass. 30,– DM! Hefte für alle Commodore! Anschrift: Christian Schmidt, Hauptstr. 62, 8737 Sulzthal

Verk. gut erhaltenen Plus4 m. 3 Joysticks, Datasette, Joy-Adapter u. vielen Spielen z.B. Strip-Poker, Paperboy, Rockman I u. II, zum Spottpreis von 200,– DM. Meldet Euch sofort unter: 04856/480

AUFGEPASST! Verkaufe Plus4, Datasette, 1 Joy, 11 Comp.-Hefte, 22 Spiele, 3 Handbücher, 6 Mon. alt, Preis nach VB od. Tausch gg. C64. Tel. 0201/466338

C16/116/Plus4. Entwicklung v. Rechenprogrammen jeder Art. Statistik u. Kalkulation! Martin G. Maaß, Kulbrockstr. 2, 4800 Bielefeld 14

Suche Tauschpartner f. C16/Plus4. Keine Anfänger! Keine Basic-Programme! Habe die neuesten Spiele u. Anwender-Programme, auch 64K! Listen an: A. Blume, Kessberg 26, 4796 Salzkotten 2, Tel. 02955/6222 (Disk. u. Tape). Bitte nach 18 Uhr anrufen!

ACHTUNG, COMPUTER-FREAKS! Verkaufe preisgünstig! Commodore VC20 u. Datasette. Preis nach Vereinbarung! Tel. 02638/1546 ab 18 Uhr.

SUCHE: Dringend billige u. gute Spiele u. Programme f. C16/116/Plus4, Kass. u. Disk. Und wer schenkt mir Computerschrott aller Art? Übernehme Porto (Bitte). Markus Tillenburg, Bilker-Allee 219, 4000 Düsseldorf 1

ACHTUNG! C16/116/Plus4/C64!! Ich suche div. Bücher zu diesen Computern. Zahle bis zu 50% vom Neupreis. Angebote: Jens Schröder, Osterholzer Möhlendamm 55,

VERKAUFE C64, Floppy, Drucker, Farbmon., Joystick u. div. Disk. inkl. GEOS. Kompl. f. 1.100,– DM VB. Bernd Reuter, Zum Schloßpark 1, 5500 Trier

Suche billige 64K-Erw. f. den C16. Bitte Angebote an: Michael Masek, A.-v.-Menzel-Str. 10, 6090 Rüsselsheim, Tel. 06142/59383

VERKAUFE PUBLIC DOMAIN-SOFT f. C16 u. Amiga. Public Domain-Soft ist frei kopierbar und daher keine Raubkopie. Liste gg. 50 Pf. Rückporto anfordern bei: Andreas Langlet, Margeritenkamp 9, 2150 Buxtehude

Tausche Software f. C16/116 u. Plus4 (auch 64K). Liste an: Mario Sailer, Hillerstr. 10, A-4030 Linz

TESTWELT sucht noch Redakteur m. C16/Plus4 u. Drucker! Mitglieder werden auch noch angenommen! Bitte schreiben (m. Rückporto!) an: TESTWELT, c/o Chr. Schweier, Hermannstr. 98, 4330 Mülheim/Ruhr. Tel. 0208/421996. Nur C16/P4!

ACHTUNG! C16/116/Plus4/C64!! Ich suche div. Bücher zu diesen Computern. Zahle bis zu 50% vom Neupreis. Angebote: Jens Schröder, Osterholzer Möhlendamm 55, 2800 Bremen 44, od. Info unter Tel. 0421/421238

Nur 5,– DM kostet Interterminal CDFÜ-Prog. (Disk./Kass.) f. C64. Text senden, mitschreiben auf Disk. Sofortausdruck, 300 – 1200 – 2400 – 4800 Baud, Passwort speichern, Disk. formatieren. K. Umtausch. M. Gutt, Kantstr. 16, 2120 Lüneburg

Tausche LCD-Spiel gg. ein Computerspiel f. meinen Commodore C64 (nur auf Steckmodul) !! EILT SEHR! Tel. 06344/3431

PLUS4-FREAK sucht Kontakte zu Club od. Anwendern. Dr. Ralf Uebel, Raschwitzer Str. 3, DDR-7113 Markkleeburg

SUCHE gute Software f. P4, auch gute Adressen f. Kauf – Tips – Kontakt. Angebote an: Andreas Schmid, Löwenstr. 62, 7000 Stuttgart 70

MAKE A MILLION! Jetzt kann jeder beweisen, daß in ihm das Zeug zum Börsenmakler steckt. Am großen Postspiel kann jeder teilnehmen! ALSO LOS! Infos gibt es gg. Rückumschlag bei: Müller, Moorweg 11, 2875 Ganderkesee

C16/PLUS4! Verk. Script/Plus-Modul m. Handbuch f. 30,– DM. W. Meier, Parkstr. 18, 2940 Wilhelmshaven

PLUS4-PROFIS GESUCHT! Neugründung! Endlich gibt es einen Profi-Club! (Nur Disk) Clubzeitung, Infoaustausch etc. Manuel Surmann, Auf dem Hagel 9, 5063, Overath, Tel. 02206/2452

VERKAUFE: C116/C16/Plus4. Anschlußkarte f. Tastatur m. 10er Block (CBM 600/700), Erw. Karte m. 2 Steckplätzen f. Expans.-Port, z.B. f. Eproms. Brenne Ihre Eproms. Suche Hard- u. Software aller Art. Tel. 040/7125694

Verkaufe meine Software f. 100,– DM, sind ca. 300 Prog. Bin auch an Tausch interessiert (Disk). Angebote od. Listen an: Bernhard Bierek, Am Unteren Weinberg 5, 8403 Oberndorf, Tel. 09405/3574

SUCHE FÜR PLUS4 gute Anwend.-Prog. wie z.B. Superbase u. Turbo-Plus-Modul. Verkaufe MPS 801 m. Ersatzfarbband f. DM 180,–. R. Graf, Ingeborgstr. 28, 5600 Wuppertal 2, Tel. 0202/555892

DDR-Student sucht f. C64 Floppy 1541 (1571) billig od. geschenkt (auch Interesse an Final Cartridge, CP/M, Disk.). Andreas Schölz, Max-Planck-Ring 4/408, DDR-6300 Ilmenau

Programmiere spezielle Anwendungen f. C16 u. Plus4 (z. B. f. Sportclubs usw.). Info bei: Christian Picker, Emsgrund 11, 4410 Warendorf 1. Preis VB. Tel. 02681/1784

C16, 128K-Erw., Floppy 1551, Datasette 1531, 2 Joyad., Bücher, Sonderhefte, viele Orig.-Spiele u. -Prog., Turbo-Tape, Text-Manager, Dateimanager, SYS-Cracker, Disketten: FP 500,– DM. 64K-Erw. (Steckmodul) 40,– DM. Möglichst Selbstabholer. Günter Sczepannek, Dobergstr. 79, 4980 Bünde

VERKAUFE Public Domain Soft f. C128. Je Disk. nur 10,– DM! Info gg. Rückporto von: M. Hackstein, Paul-Schossier-Weg 4, 4650 Gelsenkirchen

Plus4/C16/C116: Biete 50 Spiele, div. Anwend.-Prog. f. DM 20,– (Disk). Schein od. Scheck. Suche Kopierprog. v. Kass. auf Disk. Listings gg. 80 Pf. in Briefmarken.

VERKAUFE: Multiprommer V 1.5, neu, VB 120,– DM; Pulsar-Eprommer, def., VB 50,– DM; Final Cart. VB 40,– DM; Simons Basic-Modul VB 40,– DM; DB-Bücher: „Cass.-Buch“ 15,– DM; „Trainingsbuch f. Simons Basic“ 30,– DM; und jede Menge Orig.-Prog. F. Diesel, Oberländerstr. 19, 8400 Regensburg

Habt Ihr Computerschrott? Na, dann nicht wegschmeißen, sondern an: Frank Bauer, Seeor 2, 8261 Polling schikken. Zahle bis 5,– DM! Evtl. auch mehr. Habe auch eine Menge Prog. zum Tauschen. Nur Disk!

Verk. Plus4 m. Joystickadapter, 2 Handbüchern, Datasette 1531 u. 16 Spielen f. 160,– DM! Patrick Kontecke, Murjahnstr. 11, 2058 Lauenburg, Tel. 04153/3852

C16/Plus4. Suchst Du spez. Software auf Disk? Riesenauswahl! Tausch od. Verkauf mögl. Suche Lightpen(Trojan) auch o. Softw. Info bei: Frank Burzlaff, 2120 Lüneburg, Tel. 04131/50421 nach 18 Uhr

---

Suche Datasetten f. C16. Tel. 0221/525471 ab 18 Uhr

SUCHE**FLOPPY 1541** Bitte anrufen, wenn Sie ein gut erhaltenes u. billiges Gerät haben! Tel. 06121/420450, Holger Kiefer, Dieselstr. 24, 6200 Wiesbaden. VERKAUFE Orig.-Spiele (Kass.) wie Uchi Mata, Titanic usw., 10 bis 15 DM.

C128*C128*C128: SUCHE Anwend.-Prog. im C128- u. CP/M-Modus. Möglichst günstig! Gunnar Hoenekop, Asternweg 2, 4792 Bad Lippspringe

---

PLUS4: SUCHE KONTAKT zu Script/Plus-Benutzern. Peter Hakenjos, Werderstr. 13, 7500 Karlsruhe

---

*** COMMODORE PLUS/4 Wer möchte am P4-User-Port eine Telefonwählautomatik u. Datenbank (private Tel.-Anlage) betreiben? Ich helfe gerne beim Bauen. Info u. Listing bei E. Sirges, Bismarckstr. 37, 5100 Aachen, Tel. 0241/513765 (bitte Rückporto!)

BSC sucht Dich! Plagt Dich die Langeweile? Dann bist Du an der richtigen Adresse ! ! ! Hast Du Topgames, tauschen wir gern mit Dir, wenn nicht, ist es auch egal. ASCII 80/76/75/67/49/48/53/56/57/54/ 7530 Pforzheim od. Tel. 07239/7367

VERKAUFE Orgel Yamaha 35 N, 2-manualig, Rhythmus, Baßläufe, 1-Finger-Automatik u.v.m. NP über 3.000,– DM, jetzt nur 1.350,– DM od. Tausch gg. Amiga 500 m. Mon. 1081. Tel. 0711/846644

VERKAUFE f. C16/Plus4 Spiele u. Anwenderprog., 20 Prog. zu 10,– DM. VERKAUFE C16-Orig.-Spiele zum halben Preis z.B. P.O.D, R16, Attack. Schreibt an: C. Schnoor, Lausanner Str. 129, 2800 Bremen 44

HALLO, FREAKS! Ich suche die Spiele: Airwolf, Rockman, Vegas, Jackpot, Hustler zu einem billigen Preis. Suche auch den Drucker Okimate 20, er muß funkt.-fähig sein. Nicht über 300,– DM. Schnell anrufen unter 04838/965, Lars Metz

---

Plus4/C16/116. TAPE. Du, ja genau Du! Habe billige Anwender, Spiele, Grafiken. Karte od. Brief an: H.-H. Boß, Hauptstr., 2992 Dörpen

---

SUCHE TAUSCHPARTNER f. C16/C64! C16 auf Tape, C64 auf Disk. Neuester Stuff. Listen an: M. Esser, Schützenstr. 2, 4980 Bünde 1

VERKAUFE Orig.-Spiele auf Kass. z.B. Quiwi, Paint-Box. Winterolympiade, Musik-Master ab 5,– DM. Liste anfordern bei: Burkhard Koss, Theodor-Storm-Str. 16a, 2215 Hademarschen

---

HILFE HELP Plus4/C16/116: Wer hat aus dem RUN-Sonderheft das Spiel „Das Schwert der Samurai" eingegeben und zum Laufen gebracht? Portokosten werden übernommen: Tel. ab 20 Uhr 04421/25982

Wer bringt Basic-Programme v. CBM (Floppy 4040) auf IBM-Format od. kann Hinweise u. Tips geben. Gräfe, Pillauer Str. 2 G, 2804 Lilienthal, Tel. 04298/3943

EILT: SUCHE C64 u. 1541. Geräte müssen 100% o.k. sein. Zahle f. beide Geräte bis 350,– DM. Sendet Euer Angebot an: Uwe Ruettgers, Opderbeckweg 18, 4770 Soest, Tel. 02921/82118

Data-Club informiert. Data-Club! Der neue, brandheiße, druckfrische Data-Club-Katalog ist da ! ! F. nur 80 Pf. erhalten Sie ihn! Software am laufenden Band! Data-Club, Kurze Str. 3a, 2993 Esterwegen

---

Suche f. C16 (64K) Anschluß f. RGB-Monitor u. Anwenderprog. auf Eprom, Modulsoftwareplatinen u. Eprommer sowie Artikel aus CW, C64, Compute mit (keine Spiele) Tel. 02861/2994, tägl. ab 17 Uhr

---

Biete Plus4 m. 2 Floppies 1551 u. Drucker MPS 801 wg. Systemwechsel (PC) als Kompl.-Paket an für 999,– DM. Edwin Oestreich, Am Rain 11, 6423 Wartenberg 1

C16/Plus4: Verk. digital. Grafiken u. Sound-Demos! So gut wie auf C64 od. besser!? 8 Disk. Info bei: Tel. 02955/6222 – Paderborn

---

**** SDS V 2.3 **** DAS Zeichenprog. Versch. Zeichensätze, sehr schnell, viele Funktionen wie z.B. Shapes, Lupe. Zeichensatzeditor u. universelles Ausdruckprog. werden mitgeliefert. Zu bestellen gg. 20-DM-Schein bei: Holger Scherer, Jungstr. 25, 6550 Bad Kreuznach

---

SUCHE: C64 bis 150,– DM; 1541 bis 200,– DM; 1571 bis 280,– DM; PC 10 II bis 1.000,– DM; Amiga 500 – 2000. Tel. 0202/469243

Verkaufe f. C64 preisgünst. Software. 50 Games f. nur 100,– DM (nur auf Kass.). Schein an: Michael A. Schön, Landgraf-Philipp-Platz 10, 3559 Haina/Kloster

---

SUCHE FÜR PLUS4 Floppy 1551 u. Drucker Seikosha SP 180 VC od. kompatibel sowie Script-Plus-Modul gebraucht. Angebote an Georg Csapo, Danziger Str. 44, 7410 Reutlingen, Tel. 07121/36794 od. tagsüber 07071/89401

Suche Listing od. Kass. Biorhythmus f. Plus4. Bei Gefallen gute Bezahlung. Tel. nur ab 19 Uhr 07673/30724-Austria

---

****
PLUS 4 C – 16
*** 80 Zeichen Textverarb. ***
*** deutsche Umlaute ***
*** Disk DM 39,90 Fa. BLK ***
*** Kaspar-Spät-Str. 15 ***
*** 8000 München 90, ***
*** Telefon 089 / 68 82 26 ***
****

EILT! Suche C128D od. 128 u. 1571. Geräte müssen o.k. sein. Sendet Euer Angebot an: Uwe Ruettgers, Opderbeckweg 18, 4770 Soest

---

Ich besitze einen Commodore 128 u. suche verzweifelt nach einer Textverarb., die dem „Loco Script" des Schneider Joyce nicht nachsteht. Wer kann mir helfen? Zahle auch. Suche Anl. f. Superbase u. Protext. Oliver Tieste, Im Grunde 20, 2804 Lilienthal

---

Suche Speichererw. f. C128D auf 512 KB. Angebote an: Joachim Junige, Treschkowstr. 34/6, 2800 Bremen 1, Tel. 0421/72606

SUCHE COMPUTERCLUB (NUR PLUS4) Schickt mir Informationen od. ein Probeheft einer Clubzeitschr., falls vorhanden. TAUSCHE, SUCHE U. VERKAUFE Top-Software (habe ca. 300). Tel. 07191/67781 (Atze). Danke! A. Keller, Drosselweg 25, 7150 Backnang

---

C16/116/Plus4/64!!! Wir suchen noch Mitglieder. Wir bieten: Clubzeitung, Softwaretausch, Anzeigenservice und, und, und. Info gg. Rückporto. Suche Tauschpartner f. Plus4. T. Esser, Schützenstr. 2, 4980 Bünde 1, Tel. 05223/15345

SOFTWARE-PROBLEME??? Sind Sie Besitzer eines Commodore 16/116/Plus4 u. suchen Orig.-Software u. Zubehör zu niedr. Preisen? Fordern Sie einen Gratis-Katalog an! P. Schäfers, Riekestr. 5, 4402 Greven 1

VERKAUFE C16 (defekt), Adapter, Verbindungskabel, Bedienungshandbuch, Data-Becker-Buch, 42 Orig.-Spiele u. Anwend. Prog., Joyst., Commodore Welt u. Sonderhefte. Preis 210,– DM. M. Greifenhagen, Stöckelstr. 8, CH-8610 Uster

Wer verschenkt an armen Schüler einen Commodore C64. Wir brauchen ihn zwecks Schule. Wir zahlen höchstens 50,- DM. M. Greifenhagen, Stöcklerstr. 8, CH-8610 Uster, Schweiz

---

C16/64 KB m. Datasette 1531 u. 50 Spielen, Ace, Ace 2, Summer Event u.a. 200,- DM; C16 m. 1531 f. 150,- DM. Jörg Hildebrandt, Schlenkenbrink 25, 4973 Vlotho-Uffeln

---

Freesoftw. f. C16/116/Plus4: 3600 Block = 6 Diskseiten nur 20,- DM. Über 100 Programme. Viktor Jürgens, Prinzenstr. 131, 2330 Eckernförde

---

Ich suche START u. SYS-Adressen f. den Plus4. Suche DINA4 Hardcopy f. Plus4 u. MPS 100. Tel. 04192/4726, Thorsten Idzellis, Brambusch 23, 2357 Bad Bramstadt

---

Verkaufe 35 Plus4-Spiele od. Anwender auf Disk. f. 20,- DM. Lieferung nur gg. Vorkasse. Frank. Rückumschlag m. 20,- DM an: Stefan Hoffmann, Gasstr. 17, 2308 Preetz. PS: Tausche Programme

---

C16/116/PLUS4! Wer hat gute Programme (abgetippt, gekauft usw.) u. möchte mit mir tauschen? Jeder Brief wird beantwortet!! Schreibt an: Christian Bartsch, Norkshäuschen 136, 5600 Wuppertal 1

Achtung! Verkaufe Commodore Monitor 1702, techn. wie neu, m. Verp. Anschluß an C64, Preis 500,; DM VB. Tel. 06024/7939. Möglichst im Raum Aschaffenburg!

HILFE HILFE HILFE!
Wer kann mir helfen? Suche jemand, der mir bei der Basic-Sprache f. den C64 weiterhilft. Bitte melden! Norbert Korthols, Bahnhofstr. 185 A, 4100 Duisburg 12, Tel. 0203/439513

Suche f. meinen Neffen(12 J.) Floppy u. Programme f. Plus4. Gerhard Andermann, An der Hofbreite 2, 3412 Nörten-Hardenberg, OT Elvese, Tel. 05503/2590 nach 19 Uhr

---

Suche C64 od C128 m. Floppy, Drucker, evtl. auch Monitor. Nehme auch Einzelteile, wenn günstig. Tel. 069/302940.

Verkaufe C16/64K m. Floppy 1551 u. Büchern gg. Höchstgebot. Tel. 02272/7942, Heinz Lurz, Schützendelle 63, 5012 Bedburg

---

PLUS4/TEXTADVENTURES: Erfahrungsaustausch u. evtl. gemeinsames Entwickeln v. Programmen wünscht Wolfgang Dethlefs (39 J.), Krokusweg 7, 2382 Kropp, Tel. 04624/1760

Verkaufe Spiele f. C16, Orig. auf Tape zu niedr. Preisen. Ruft an 06233/62937. Andreas Fehres. Jede nur einmal. Grüße M. Deis. Schreibe mal wieder!

---

60 Spiele f. C64 f. 30,– DM sowie C16/116/Plus4 ebenfalls 30,– DM. Schick die Scheine an: Alexander Käss, Himmelstr. 7, 7987 Weingarten und Ihr bekommt die Software.

C16(64K), Datasette, Floppy 1551, M&T-Sonderhefte, Prog.-Service-Disketten, Spiele auf Disk. u. Kass. (keine abgetippten!), Textverarb. u. Dateiverw. Alles 1A-Zustand, (auch einzeln) billig abzugeben. Tel. 02821/20050 ab 18 Uhr.

Suche Textmanager m. Such- u. Sortier-Routine, Plus4, Kass. Jürgen Schust, Försterweg 40, 2000 Hamburg 54, 040/5407326

C16, 116, Plus/4-Anwender: Der Verein speziell f. uns! Mit Clubzeitschr. u. Software-Bibliothek. Info beim hrc e.V., Bauerland 15, 48 Bielefeld 1

Hallo, Plus/4-C16-Fans! Plus4-User aus der DDR sucht Erfahrungsaustausch u. Briefwechsel. Bin 31 J. (M). würde mich auch sehr über Soft- u. Hardware freuen, da mittellos. Jeder Brief wird beantw. v. Frank Harnisch, Ringstr. 189, DDR-7060 Leipzig

---

Verkaufe versch. Zeitschr. (64er, RUN, CINEMA ...) sowie div. Orig.-Prog. (Tape u. Disk.) f. C64. Liste gg. Rückporto. Kaufmann, Burgstr. 61, 7000 Stuttgart 80

Suche intakte Floppy 1551 u. GRAFIKMODUL f. den MPS-802/1526 od. Plotter!! Angebote an: Brigitte Nitsch, Steinbruch Str. 17, 4100 Duisburg 1, T. 0203/358074. PS.: Tausche auch Top-Soft auf Disk.! Beantworte alles!

Für 10,– DM bekommt Ihr die besten Spiele für den C16/+4: 2 Disk. randvoll m. Topsoftware! Suche noch Partner zum Software-Tausch. Nur Disk. Suche noch Superbase (64K). Bitte wenden an: A. Balgar, Fortkamp 7, 4250 Bottrop

---

Für Plus4, C16(64K): Mathimago 30,– DM; Portfolios 15,– DM. Info bei: Peter Hakenjos, Werderstraße 13, 7500 Karlsruhe

---

Suche Grundplatine f. C16 (nicht defekt). Angebote an: Lothar Härterich, Hauptstr. 85, 8721 Poppenhausen, Tel. 09725/1434 ab 18 Uhr

---

Hey, Computer-Freaks! Ich suche div. Data Becker Bücher f. C64/C16/P4. Zahle bis zum halben Neupreis! Schickt Eure Angebote an: Jens Schröder, Osterholzer Möhlendamm 55, 28 Bremen 44, Tel. 0421/421238

Verkaufe meinen C16 + 64 KB. Mit Datasette, Basic-Kurs usw. Ebtl. Plus4 m. 256 KB. Preis VB. Verk. auch einige Orig.-Software. Axel Kleinjung, Dümmlinghauserstr. 22, 5270 Gummersbach 1, Tel. 02261/52644 v. 16-17 Uhr

---

Verkaufe DFÜ-Prog. u. Taschenrechnerfunktionen. Disk.: 10,– DM. ACHTUNG: 1. Prog. arbeitet nicht mit allen Dataphonen. 2. Keine Texteingabefunktion. Prog. f. C64. Kein Umtausch. Matthias Gutt, Kantstr. 16, 2120 Lüneburg

---

Suche Footballer of the Year f. Plus4. Der erste erhält 2 Orig.-Spiele (Tape). Markus Ramm, Frankenweg 33, 4630 Bochum 6

Achtung! Suche Drucker f. C64. Da total verarmt, kann ich nur bis zu 150,– DM zahlen! Tel. 02392/71589, Raum Werdohl. Dringend !!!

C16/+4 Gratis-Info anfordern. Kopiermodul, Kopier-Prog., Anwender- u. Spiel-Prog. Jetzt Gratis-Info anfordern!! Th. Görtz, Friedrich-Ebert-Str. 113, 6103 Griesheim, Tel. 06151/719132

ACHTUNG! C16/116/Plus4. Mit POKE4097,1:DELETE1 macht man Reset und new rückgängig. Suche C64 bis 150,– DM, kein Schrott. Tel. 04861/6264. Ich rufe zurück.

IDEAL FÜR EINSTEIGER! VC20 m.: Datasette, 3K-Erw. 3 Spielemodulen u. 17 Kass. (4 neue, 5 eigene, 7 Orig.-Spiele, Basic-Kurs). M. Porto nur 100,– DM. Köneke, Westendstr. 26, 6072 Dreieich

---

C128D, neuestes Geräte (Stahlgehäuse), 80-Z.-S/W-Mon., 70 Disk. u. div. Zubehör, alles 6 Mon. alt. VB 900,– DM. Tel. 02234/58964 ab 18 Uhr

---

Input-Kass. 3/85-8/87 f. 150,– DM; Profimat m. Disk. f. 50,– DM; Roboter-Buch 20,– DM; Lexikon C64 25,– DM; Codier-Blätter Conrad-basic 200 St. 6,– DM. Wolfgang Bruns, Graf-Bernhard-Str. 22, 4934 Horn-Meinberg 01. Tel. 05234/1484 ab 18 Uhr

---

HEY COMPUTERFREAKS! Verk. C16 + 64K 150,– DM; Datasette 75,– DM; Floppy 400,– DM; 10 Leerdisk. 40,–; 2 Quickshot II 25,– DM; 2 Adapter 10,– DM; Sonderhefte, VIA, Software (z.B. Winter Olympiade u.v.m.). CDS, 44, rue de Mondercange, 4381 Ehlerange-Luxemburg

---

Tausche, kaufe, verkaufe Spiele, Anwend.-Prog. u. Musik-Prog. f. C16/+4. Suche 2 Joysticks z. günst. Preis! Bitte um Angebote, Listen u. Ähnliches. Jede Zuschrift wird garantiert beantwortet! Alles an: Jürgen Oschinger, Eschweg 1, 2171 Geversdorf/Oste

---

* HALLO C16/P4 FREAKS* Verkaufe 20 Prog. f. nur 10,– DM. Auf Disk.! Verk. 50 Prog. f. nur 20,– DM. Auf Disk.! Bei Interesse Schein an K. Henk, Breslauerstr. 4, 518 Eschweiler

---

An alle C16/Plus4-Fans! Bieten ca. 400 Games ab 5 Pf. pro Spiel, Liste & Lieferbedingungen bei Jordi Rieger, Maiburgerstr. 38, 2950 Leer/Logabirum anfordern od. anrufen 0491/73808, Spider-Crack Computer-Club

---

Suche: „Mein 1. C64-Programm“ v. G. Lippman, Sybex-Verlag. Riedl, Tel. 0911/440166

Verkaufe div. Input 64, Software-Club-Kassetten u. Commodore Discs (teils orig.-verpackt). Angebot an Till Franzmann, Tel. 06132/8168 (Freitag ab 19 Uhr)

C64 – C16 – Plus4 – Lernprg. Techn. Mathe + Schulanwend. + Grafik zu reellem Preis. Cass/Disk z.B. Bruchrechn. Vokab. Geometrie, Zahnr. Festigk. Hydr. E-Techn. Katalog 1 DM Briefm.-Comp. Typ angeben! Softvers. A. Ristau Peetzweg 9, 3320 Salzgitter 1

Verk. 2-Color-Printer CITIZEN IDP-560 f. Commodore-Comp. 60,– DM, inkl. Anwenderprog. f. den Drucker. Etiketten – Text – Tansparent

---

LOGO für den Plus4 od. C16 (64K). Steckmodul, Demo-Disk. u. Handbuch f. DM 50,–. Bei Vorkasse NN +5,– DM. Harald Hobbelmann, Junkernkamp 18, 2822 Schwanewede, Tel. ab 18 Uhr 04209/5390

ACHTUNG, C16-FREAKS! Verk. Prog. ab 50 Pf. 60 Prog. f. 35,– DM. Liste gratis: H.P. Baumgartner, Osloring 17, 23 Kiel, Tel. 0431/522443 ab 19 Uhr.

ACHTUNG! Suche dringend CW-Sonderheft (C16/116/P4) Nr. 3/86. Zahle gut! Anrufe zw. 17 u. 20 Uhr! Es eilt sehr. Tel. 0228/655110

FLOPPY 1551 f. C16/116/P4 m. Disketten, ca. 200 Prog. (Spiele u. Anwend.) wg. Systemwechsel f. 200,– DM. Tel. 19–21 Uhr, Thomas Kolbe, Saselerstr. 167, 2000 Hamburg 73

SUCHE KONTAKT zu C16- od. Plus4-Usern zwecks Erf.- u. Software-Tausch aus dem Raum 5800 Hagen. Kontaktadresse: H. Ritter, Berghofstr. 22, 5800 Hagen, Tel. 02331/83140

VERKAUFE C16 m. Abdeckhaube, Datasette, Spielen, Heften, Büchern f. 200,– DM Schreibt an: Andreas Fehres, Schwalbenweg 15, 6710 Frankenthal, erst o. Scheck. Schreibe dann zurück.

SUCHE dt. Anleitung f. SCRIPT PLUS Textverarb. (auch selbstgeschrieben). Gerd Besser, Bahnhofstr. 13, 5982 Neuenrade (Märk. Kreis), Tel. 02392/64587

– usw. Auf Disk. od. Kass. zus. 11 Prog. f. C16 od. P4. Hermann Rippel, Vogelsbergstr. 112, 6420 Lauerbach

---

C16/Plus4! Anwender-Software. GRATISINFO bei: Frank Murovec, Alleestr. 56, 4050 Mönchengladbach

---

C16/116/P4. Übernehme Auftragsprogrammierungen Ihres Systems. Mit Vorliebe mathematisch/naturwiss., Datenverw., Lageristik! Jürgen Vorwerk, Dörener Weg 29 a, 4790 Paderborn

---

C16-FAN sucht alles f. C16 (Lit., Soft-, Hardware) u. Kontakte im Raum Köln. Tel. 0221/705923

*** MCS 801 *** MCS 801 Suche Tips u. Tricks f. o.g. Drucker! Jede Menge Prog. zum Tausch. Frank Bauer, Seeor 2, 8261 Polling

---

Suche den, der mich aus Bochum angerufen hat u. mir eine Bezugsadresse f. SCRIPT PLUS m. dt. Anl. genannt hat. Bitte melden Sie sich noch einmal. Mir ist die Adresse abhanden gekommen! Brigitte Nitsch, Tel. 0203/358074

VERKAUFE C16 (64K) inkl. Datasette u. 12 Spiele (Kass.) u. Anwend.-Prog. DM 150,–. Marco Brockschmidt, Forstweg 7. 4817 Leopoldshöhe

---

C16/116(64K)/Plus4: Relat. Dateiverw.: bis zu 1500 Datensätze! Freie Maskenerstellung in 10 Feldern á 24 Zeichen! Mit Handbuch! Preis 19,50 DM plus NN/Porto. Jürgen Vorwerk, Dörener Weg 29a, 4790 Paderborn

---

PLUS4 zu verkaufen. Commodore Plus4, Floppy 1551, Datasette, Disk.-Box u. ca. 35 Disk., Bücher u.a. f. VB 400,– DM. Dierk Roessner, Eppendorferweg 115, 2000 Hamburg 20

VERKAUFE Zenith Z 148 PC, 100 % IBM-kompatibel m. 20 MB HD, Mannesmann Tally MT 85 u. vielen Prog., f. nur DM 4.000,–. Ab 17 Uhr unter 089/529574 od in AV-Mailbox 089/183951, Username OWAMBO

PLUS4, defekt, v. Bastler gesucht. Bitte Angabe v. Fehler u. Preisvorstellung an: Hellmut Kainka, Edm.-Weber-Str. 186, 4690 Herne 2, Tel. 02325/62556

C16/Plus4-User, verkaufe meine Software (meist Spiele). Liste gg. Freiumschlag superbillig v. Harald Scheel, Kollaustr. 178, 2000 Hamburg 61

VERKAUFE meinen C16 (64 KB), Zeitschriften, 24 Spiele (nur Orig.) f. nur 70,– DM. Bitte melden bei: Jörg Steinhilber, Am Weiher 6, 6120 Michelstadt-Asselbrunn

---

TED-Chip 8360R2 f. C16/Plus4 DM 20,– zu verkaufen. Tastatur CBM 610 DM 8,–. Sonstige Chips, Decoder, ROMs, eing. Software teilweise noch vorhanden. Stübs, Tel. 02351/6956 ab 18 Uhr

---

C16/116/Plus4 Verk./tausche günst. TOP-Programme (habe über 400 St.) u.a. Copy-Prog. T/D-D/T usw. Verk. auch C116 günstig. Tel. 07191/67781. Listen bei: A. Keller, Drosselweg 25, 7150 Backnang. Möglichst 50 Pf beilegen.

SUCHE ANWENDUNGEN im kaufmänn. Bereich sowie dt. Handbuch f. Floppy 1551. A. Jacksch, Schneiderberg 15, 3000 Hannover 1, Tel. 0511/7000039

C16/116/P4 Suchst Du Software? Ich habe sie. Liste gg. Rückporto. Frank Bauer, Seeor 2, 8261 Polling, Tel. 08631/14588, ab 17 Uhr

---

SUCHE dringend einen P4. Angebote an: Oliver Thomas, 6729 Wörth 2, Tel. 07271/41265

Checkmon

Jetzt perfekt: Unser Checksummer

Hatte bisher unser Checksummer an Buchstabenvertauschungen nichts auszusetzen, so zeigt er sich nun nicht mehr so kulant.

Ob Sie mit der alten Version nun eingegeben hatten:

10 print"ab"
oder
10 print "ba",

der Checksummer brachte in beiden Fällen die Prüfsumme < gk >. Leicht kann es vorkommen, daß beim schnellen Tippen, besonders im Zehnfingersystem, die Taste, die eigentlich erst als übernächste drankommen sollte, ein wenig zu früh erwischt wird. Dem Checksummer, der lediglich die Ascii-Werte der Buchstaben addierte, konnte dieses natürlich nicht auffallen. Was also tun? Ob etwas früher oder später addiert wird, ändert nichts am Resultat der Summe. Anders ist es, wenn man zwei Verknüpfungsarten kombiniert. So ist z.B. 2*30+40 etwas anderes als 2*40+30. Und genau dieses war dann die Lösung. Die Summe wird nun einfach durch eine Linksverschiebung vor jeder Addition verdoppelt. Dadurch, daß im Falle, wenn das Ergebnis größer als 255 ist, der dabei entstehende Übertrag als Wert 1 zusätzlich addiert wird, verflüchtigen die Werte der am Anfang der Zeile gefundenen Codes sich nicht nach 8 weiteren Zeichen. Damit bleibt nicht nur die Aussagekraft der Prüfsumme voll erhalten, sondern erfährt sogar eine erhebliche Steigerung. Und vor allen Dingen wird nur eine klitzekleine Änderung erforderlich, die dieses zu vollbringen, in der Lage ist. Ein einziges Byte ist nur zu ändern. Wir tun dieses mit "poke 345,10" in der Zeile 470. Dadurch wird das hier ursprünglich ansässige CLC (Clear Carry) durch ASL (Arithmetik Shift Left) ersetzt. Die nachfolgende Addition mit ADC (Addiere mit Carry) addiert den Ascii-Code des gefundenen Zeichens und den nach links herausgeshifteten Übertrag. Da einige unserer Leser beklagten, daß das Checksummerlisting nachher noch im Programmspeicher stehen würde, haben wir diesem noch mit einem "new" abgeholfen. New bzw. neu ist nun folgendes.

10 print"ab" ergibt die Prüfsumme< jd >
10 print"ba" die Prüfsumme<jf>

Sie brauchen den Checksummer nicht neu einzutippen. Alles, was Sie tun müssen, ist, die Zeile 470 anzufügen. An der Bedienung des Checksummers hat sich nichts geändert. Die Eingabehinweise bleiben daher wie gehabt.

EINGABEHINWEISE

Am rechten Rand jedes Listings, jeweils am Ende einer Eingabezeile, finden Sie zwei Buchstaben zwischen einem Kleiner- und einem Größerzeichen eingeschlossen. Diese dürfen Sie nicht mit in Ihr Listing eintippen, sondern sie dienen Ihnen zur Überprüfung Ihrer Eingabe.

Zwischen dem Kleiner- und dem Größerzeichen am rechten Rand befinden sich zwei Buchstaben. Mit einem speziellen Programm können Sie beim Eintippen Ihre Eingabe auf ihre Richtigkeit überprüfen. Dieses Programm, der Checksummer, sorgt nämlich dafür, daß nach erfolgter Zeileneingabe am linken oberen Bildschirmeck zwei Buch-

ERST SICHERN, DANN AUSPROBIEREN

staben ausgegeben werden. Wenn diese Buchstaben nicht mit den vorher erwähnten Buchstaben in unserem Listing übereinstimmen, so können Sie davon ausgehen, daß Sie sich vertippt haben und können sich so die Zeile nochmals näher ansehen, ob Sie Ihren Eingabefehler finden. Wenn Sie dann alles richtig getippt haben, so stimmen die Buchstaben überein und Sie können sich getrost der nächsten Zeile zuwenden.

Das Checksummerlisting hat noch keine Prüfsummen. Seien Sie deshalb besonders aufmerksam, daß alles paßt und speichern Sie dieses Programm unbedingt ab, **bevor Sie es starten!** Bei einem Tippfehler würde es sich wahrscheinlich auf Nimmerwiedersehen verabschieden und Sie müßten die ganze Arbeit vermutlich nochmals machen. Wenn Sie es gestartet haben, so geschieht nichts Besonderes. Der Computer meldet sich einfach kurz darauf mit „READY", und das war auch schon alles. Alles sollte nun wie immer funktionieren, mit der kleinen Ausnahme, daß nunmehr nach jeder Eingabe im Direktmodus eine Prüfsumme erscheint. Nehmen Sie zum Testen irgendeine kurze Basiczeile aus unserem Heft her und testen sie aus. Wenn die Summen übereinstimmen, so können Sie sich freuen, denn Fehler beim Abtippen werden Ihnen nun in Zukunft viel weniger passieren, als vorher.

EINER FÜR ALLE, EIN ECHTES UNIVERSAL-PROGRAMM

Unseren Checksummer können Sie verwenden, ob Sie einen C16/116/Plus4 oder ob Sie einen C64 oder gar einen C128 haben. Nur müssen Sie beim letzteren beachten, ob Sie auch wirklich im 40-Zeichenmodus sind. Nachdem Sie den Checksummer geladen und gestartet haben, können Sie Ihr Basicprogramm eingeben wie gewohnt, Sie können es abspeichern, Sie können auch laden, Sie können Kürzel verwenden und, ob Sie ein paar Leerzeichen mehr oder weniger verwenden, der Checksummer läßt sich dadurch nicht aus der Fassung bringen. Ein bißchen Vorsicht sollte man allerdings walten lassen, wenn man Programme eingetippt hat, in denen Peeks und Pokes vorkommen. Es wird zwar nicht besonders häufig vorkommen, aber es könnte bisweilen ge-

```
10 rem =checksummer==c16 c64 c128==
20 rem (p) 05/87 commodore welt   ==
30 rem ===========================
40 rem (c) alfons mittelmeyer     ==
50 rem                            ==
60 rem c16/116/plus4              ==
70 rem c64                        ==
80 rem c128 (40-zeichen)          ==
90 rem ===========================
100 rem --------------------------
110 rem   grundroutine (c16)
120 rem --------------------------
130 data165,059,072,165,060,072,032
140 data086,137,104,133,060,104,133
150 data059,152,072,160,000,165,020
160 data024,101,021,170,024,144,011
170 data201,032,240,006,138,024,113
180 data059,234,170,200,177,059,234
190 data208,240,169,031,072,138,074
200 data074,074,074,072,138,041,015
205 data072,169,031,072,162,003,104
210 data024,105,129,157,000,012,202
220 data016,246,104,168,096
230 lt=peek(772):ht=peek(773)
240 fori=312to386:readx:pokei,x:nex
t
250 iflt<>124then350
260 rem --------------------------
270 rem         anpassung c64
280 rem --------------------------
290 fori=312to317:pokei,234:next
300 fori=321to326:pokei,234:next
310 fori=1to6:readad:readx:pokead,x
:next
320 poke380,4:poke319,lt:poke320,ht
:goto430
330 data346,121,347,000,348,002
340 data351,185,352,000,353,002
350 iflt<>13then430
360 rem --------------------------
370 rem anpassung c128 (40 zeichen)
380 rem --------------------------
390 restore410:poke332,22
400 poke335,23:goto310
410 data313,061,316,062,323,062
420 data326,061,347,061,352,061
430 poke772,056:poke 773,1
440 rem --------------------------
450 rem ergaenzung 10/87
460 rem --------------------------
470 poke 345,10:new
480 rem ==========================
490 rem = fuer hefte cw 7/87 bis  =
500 rem = cw 9/87 sowie cw128 5/87=
510 rem = und c16 6/87 ist die    =
520 rem = poke-anweisung in zeile =
530 rem = 470 wegzulassen         =
540 rem ==========================
```

schehen, daß nach dem Laufenlassen eines Programmes weder der Checksummer noch sonst etwas mehr funktioniert, auch wenn dies bisher ohne Checksummer nicht der Fall gewesen sein sollte. Also bitte sichern Sie in jedem Falle Ihre Programme, bevor Sie sie ausprobieren.

Ein paar Dinge sollten Sie noch wissen. Wir drucken in unseren Listings des öfteren Punkte statt Leerzeichen. Wenn Ihnen nun aber Leerzeichen besser gefallen, so liefert der Checksummer natürlich eine falsche Summe. Wenn Sie diese Richtigkeit überprüfen wollen, so können Sie dies tun, indem Sie sie zuerst einmal so wie im Heft abtippen, und nachher, nachdem Sie sie nachgeprüft haben, einfach wieder die Punkte durch Leerzeichen ersetzen.

A. Mittelmeyer

MONITOR

CHECKMON

```
40 rem checkmon ==============c16 <cn>
50 rem (p) commodore welt team    <ke>
60 rem ========================== <nk>
70 rem (c) by  a.mittelmeyer      <ag>
80 rem c16/116/plus4              <ki>
90 rem ========================== <jg>
100 fori=312to398:reada           <ei>
110 pokei,a:next                  <ep>
120 data 132,218,108,219,0,132,219 <oe>
130 data 164,218,76,75,236,201,62 <nk>
140 data 208,249,165,161,10,101   <jc>
150 data 162,160,7,10,113,161,136 <ej>
160 data 16,250,133,216,169,30,133 <oh>
170 data 217,169,62,160,97,208,220 <mk>
180 data 198,217,208,218,160,105  <ai>
190 data 208,212,201,13,240,4,164 <ha>
200 data 218,24,96,169,60,160,68  <lh>
210 data 32,61,1,165,216,32,16,251 <ec>
220 data 169,62,160,5,208,2,169,32 <om>
230 data 32,75,236,136,208,248,169 <ol>
240 data 13,208,176,219,68,220,1  <go>
250 data 804,56,805,1             <hn>
260 fori=1to4:reada:readb:pokea,b <lm>
270 next:new                      <ji>
280 rem ========e=n=d=e========== <cc>
```

"CHECKMON" ist eine unerlaessliche Hilfe zur Eingabe von Maschinenprogrammen. Laden und starten Sie "Checkmon" und gehen dann mittels MONITOR in denselben. Wenn Sie sich nun z.B. mit 'M1000' einen Speicherbereich ansehen, oder Hexzahlen eingeben, so erscheint rechts die Pruefsumme anstatt der Asciicodes.

Werden Sie Manager!

Die Gesetze des Marktes und nicht die des Zufalls bestimmen dieses Spiel. Preisschwankungen, Streiks und Sonstiges kommen nicht von ungefähr. In einer tiefgreifenden Wirtschaftssimulation erleben Sie, daß der Beruf des Managers kein Zuckerschlecken ist.

Wer wäre nicht gerne einmal Präsident einer eigenen Firma? Dieses Programm gibt Ihnen die Möglichkeit dazu: Mit 25000 Dollar Startkapital und einer leeren Fabrikhalle fangen Sie an und müssen versuchen, mit dem Verkauf von Fertigwaren einen möglichst hohen Gewinn zu erwirtschaften. Doch der Weg zum großen Geld ist nicht leicht: Preisschwankungen, die Gewerkschaft und die Konkurrenz machen Ihnen das Leben schwer.

Nach dem Start des Ladeprogramms gelangen Sie, sobald das Hauptprogramm nachgeladen ist, ins Titelbild. Sie können mit dem Joystick wählen, ob Sie ein altes Spiel fortsetzen oder ein neues beginnen wollen. Steuern Sie den kleinen Pfeil in das entsprechende Kästchen am Bildschirm und drücken Sie die Feuertaste.

Alle anderen Abfragen im Programm funktionieren genauso, nur zur Eingabe von Namen wird die Tastatur verwendet.

Wenn Sie auf die Frage „Altes Spiel fortsetzen?“ mit NEIN geantwortet haben, müssen Sie die Spieldauer, Ihren Namen, den Namen Ihrer Firma sowie ein dreistelliges Firmenkürzel eingeben. Jetzt kann's losgehen:

Per Fernschreiber erhalten Sie wichtige Informationen wie die An- und Verkaufspreise von Rohstoffen, beziehungsweise Fertigwaren. Auch über bestimmte Ereignisse werden Sie hier unterrichtet.

Durch Druck auf den Feuerknopf gelangen Sie zum nächsten Programmpunkt:

Sie erhalten eine Auflistung Ihrer Vermögenswerte und können sich – kostenpflichtige – Informationen, etwa über die Ausgaben der Konkurrenz für Werbung und Forschung oder über den Rationalisierungsgrad aller sechs Unternehmen beschaffen.

Beim folgenden Punkt können Sie, sofern Sie Maschinen, Arbeiter und Rohstoffe besitzen, Fertigwaren produzieren. Diese sind allerdings immer erst in der darauffolgenden Runde fertig.

Jetzt können Sie Rohstoffe einkaufen und Fertigwaren verkaufen.

Geben Sie dabei die Anzahl der Waren sowie den Preis, den Sie zahlen beziehungsweise fordern, ein. Der Markt ist begrenzt und der Computer nimmt nur die besten Angebote an; Sie könnten also leicht leer ausgehen.

Die Zahleneingabe geschieht wieder mit dem Joystick, indem Sie die Kästchen mit „Plus“ beziehungsweise „Minus“ so oft anklicken, bis die Zahl richtig eingestellt ist. Daraufhin bestätigen Sie mit dem mittleren Kästchen.

Anschließend können Sie Maschinen und Computer einkaufen beziehungsweise verkaufen sowie Arbeitskräfte einstellen oder entlassen. Durch den Computer sinken die Produktionskosten um die Hälfte.

Im nächsten Menüpunkt werden die Investitionen für Werbung und Forschung, die Lohnhöhen und die Ausgaben für den Exklusiv-Vertrieb festgesetzt.

Werbung bringt Ihnen einen Wettbewerbsvorteil beim Verkauf, Forschung kann zu einer Senkung der Produktionskosten, die anfangs bei 1000 Dollar liegen, führen. Über Forschungserfolge werden Sie per Telex informiert.

Die Vertriebskosten für eine Ware betragen 500 Dollar, dafür können Sie sie jedoch konkurrenzlos absetzen.

Am Ende des Monats werden Bilanz gezogen und die Steuern gezahlt. Die Besteuerung ist progressiv, Investitionen lassen sich absetzen. Außerdem können Sie sich per Menü über die Gewinne der fünf Konkurrenzunternehmen sowie über Ihre Fabrikausstattung (Inventur) informieren. Hier kann auch das Spiel abgespeichert oder vorzeitig beendet werden. Mit WEITER beginnt der nächste Monat.

Wenn die Spielzeit abgelaufen ist, wird der Computer Ihre Fähigkeiten als Manager beurteilen. Dabei wird auch die durchschnittliche Zeit, die Sie für eine Runde benötigt haben, berücksichtigt, denn vor allem die Fähigkeit, schnell und richtig zu entscheiden, macht einen guten Manager aus.

Bei Manager handelt es sich um eine gründliche Wirtschaftssimulation. Gewinne, Einkäufe und Ausgaben der Konkurrenzfirmen werden nicht zufällig ermittelt, sondern der Computer übernimmt die Leitung dieser Betriebe. Auch die Preisschwankungen folgen den Gesetzen des Marktes und nicht denen des Zufallsgenerators. Streiks kommen nicht von ungefähr. Die Gewerkschaft verwaltet einen Streikfonds, aus dem die Arbeiter bezahlt werden. Details wie das Image der Firma werden ebenfalls berücksichtigt.

Das Programm besteht aus zwei Teilen: dem Ladeprogramm mit den MS-Unterroutinen, die den Zeichensatz enthalten, und dem Hauptprogramm. Das Nachladen erfolgt automatisch. Der MS-Teil verwaltet auch die Pseudo-Softsprites sowie die Windows. Zu diesem Zweck wurde das BASIC-Ende ($37) heruntergesetzt.

Die einzelnen Startadressen der Routinen werden am Programmbeginn Variablen (U0 bis U9) zugewiesen, ihre Funktion ist dort dokumentiert. □

```
10 rem manager =================p4 <ig>
20 rem (p) commodore welt team    <ho>
30 rem ========================== <ng>
40 rem (c) by bernhard oemer      <jl>
50 rem                            <pd>
60 rem                            <ah>
70 rem basic v3.5                 <nl>
80 rem plus4 (c16/116 + 64kb)     <fd>
90 rem ========================== <jg>
100 :                             <ac>
110 color1,2                      <gn>
120 scnclr                        <ni>
130 :                             <dp>
140 rem  unterprogramme           <af>
150 :                             <gh>
160 u0=4334 :rem sprite           <jd>
170 u1=4577 :rem delete           <pf>
180 u2=4773 :rem fast             <hm>
190 u3=4782 :rem slow             <mc>
200 u4=4791 :rem sleep            <hn>
210 u5=4863 :rem screen           <fa>
220 u6=5011 :rem move             <bn>
230 u7=5161 :rem test             <ho>
240 u8=5234 :rem wait             <ah>
250 u9=5257 :rem frame            <im>
260 :                             <ee>
270 sysu2                         <mp>
280 :                             <gm>
290 rem  variablendefinition      <cj>
300 :                             <je>
310 ho$=chr$(19)  :rem home       <hp>
320 cl$=chr$(147):rem clear       <kf>
330 up$=chr$(145):rem up          <pa>
340 dw$=chr$(17)  :rem down       <ok>
350 le$=chr$(157):rem left        <ka>
360 ri$=chr$(29)  :rem right      <en>
370 ec$=chr$(27)  :rem escape     <je>
380 rv$=chr$(18)  :rem revers on  <kl>
390 ro$=chr$(146):rem revers off  <mm>
400 rt$=chr$(13)  :rem return     <ma>
410 bk$=chr$(144):rem black       <gg>
420 wt$=chr$(5)   :rem white      <dk>
430 de$=chr$(20)  :rem delete     <hc>
440 :                             <kn>
450 ul$=rv$                       <ii>
460 fori=1to40                    <ap>
470 li$=li$+chr$(179)             <kb>
480 ul$=ul$+chr$(185)             <hg>
490 next                          <fj>
500 zz$="abcdefghijklmnopqrstuvwxy
z-&. 1234567890"                  <nh>
510 :                             <dj>
520 rw$=chr$(163):fw$=chr$(164)   <eg>
530 fori=1to15                    <gk>
540 rp$=rp$+rw$:fp$=fp$+fw$       <pp>
550 next                          <nc>
560 ab$=chr$(165)+le$+dw$+chr$(166
)+up$                             <dm>
570 cp$=chr$(127)+chr$(160)+le$+le
$+dw$+chr$(161)+chr$(162)+up$     <jd>
580 fori=0to5                     <po>
590 g1$=g1$+chr$(167+i)           <od>
600 g2$=g2$+chr$(173+i)           <no>
610 next                          <ek>
620 dimma$(2)                     <id>
630 fori=0to2                     <hk>
640 ma$(2-i)=chr$(115+i*4)+chr$(11
6+i*4)+le$+le$+dw$+chr$(117+i*4)+c
hr$(118+i*4)+up$                  <ih>
650 next                          <jl>
660 :                             <gh>
670 dimgr$(7)                     <ep>
680 fori=179to185                 <pj>
690 gr$(i-178)=chr$(i)+chr$(i)+chr
$(i)                              <fm>
700 next                          <pp>
710 :                             <ml>
720 dimi,x,y,z,a,b,c,d,e,a$,b$,c$ <dg>
730 dimna$(5),fb$(5),rw(5),fw(5),m
a(5,2),cp(5),ab(5),gd(5),sc(5),lh(
5),pk(5),ia(5)                    <nf>
740 dimik(5),ig(5),vg(5),sk(5),gw(
5),we(5),fo(5),e1(5),e2(5),e3(5),e
4(5),x(5),y(5)                    <md>
750 dimrw,fw,al,dg,da,dp,dl,dd,dk,
dm,mo,f1,f2,f3                    <kh>
760 rw%=32:rw=25:fw%=32:fw=24:am=2
0:rp=450:fp=4000                  <bg>
770 kp=5000:az=5                  <eo>
780 :                             <fi>
790 rem  titelbild                <mi>
800 :                             <ia>
810 printec$"n"ec$"m"chr$(142)chr$
(8)                               <je>
820 sysu9,0,0,39,24,2             <hf>
830 print                         <bl>
840 print" $   $  $$  $  $  $$
$$  $$$ $$$ "                     <lk>
850 print" $$ $$ $$$$ $  $ $$$$ $
$$$ $$$ $$$$"                     <hg>
860 print" $$$$$ $  $ $$ $ $  $ $
    $   $  $"                     <ld>
870 print" $ $ $ $$$$ $$$$ $$$$ $
 $$ $$$ $$$ "                     <ai>
880 print" $    $ $  $ $ $$ $  $ $
  $ $   $  $"                     <gl>
890 print" $    $ $  $ $  $ $  $ $
  $ $   $  $"                     <df>
900 print" $    $ $  $ $  $ $  $
$$  $$$ $  $"                     <ha>
910 print:print                   <aa>
920 print"   eine wirtschaftssimu
lation von"                       <ki>
930 print:print"           bernah
rd oemer"                         <jn>
940 sysu9,6,15,33,22,2            <fk>
```

```
950 a$="altes spiel fortsetzen ?":
x=26:gosub2430                         <ag>
960 sysu9,12,19,17,21,0:printrv$"
ja "                                   <ic>
970 sysu9,22,19,27,21,0:printrv$"n
ein"                                   <mn>
980 sysu0,48,120                       <fh>
990 sysu1                              <fp>
1000 sysu3                             <lb>
1010 sysu6                             <ag>
1020 sysu7,12,19,17,21,1080 :rem j
a                                      <ah>
1030 sysu7,22,19,27,21,1740 :rem n
ein                                    <ce>
1040 goto1010                          <ph>
1050 :                                 <hg>
1060 rem  laden                        <kc>
1070 :                                 <jo>
1080 sysu9,12,19,17,21,1:print" ja
"                                      <fp>
1090 sysu4,40                          <if>
1100 sysu9,6,15,33,22,2:printcl$;      <ol>
1110 a$="speichermedium ?":x=26:go
sub2430                                <eo>
1120 sysu9,12,19,17,21,0:printrv$"
tape"                                  <mn>
1130 sysu9,22,19,27,21,0:printrv$"
disk"                                  <om>
1140 sysu6                             <bk>
1150 sysu7,12,19,17,21,1180 :rem t
ape                                    <fg>
1160 sysu7,22,19,27,21,1190 :rem d
isk                                    <la>
1170 goto1140                          <fa>
1180 c=1:b=0:sysu9,12,19,17,21,1:p
rint"tape":goto1200                    <ee>
1190 c=8:b=8:sysu9,22,19,27,21,1:p
rint"disk"                             <ih>
1200 sysu4,40                          <bm>
1210 sysu9,6,15,33,22,2:printcl$;      <jf>
1220 a$="name der datei ?":x=26:go
sub2430                                <ip>
1230 print:print                       <ag>
1240 print"    >";                     <ko>
1250 x=16:gosub2270                    <kl>
1260 sysu4,20                          <bl>
1270 sysu9,6,15,33,22,2:printcl$;      <ic>
1280 printleft$(li$,26)                <ep>
1290 printrv$"legen sie die ";         <jf>
1300 ifc=1thenprint"kassette";:els
eprint"diskette";                      <gh>
1310 print" ein"                       <cc>
1320 sysu9,17,19,22,21,0:printrv$"
 ok "                                  <ci>
1330 sysu6                             <ba>
1340 sysu7,17,19,22,21,1360            <gc>
1350 goto1330                          <nk>
1360 sysu9,17,19,22,21,1:print" ok
"                                      <hh>
1370 poke154,0                         <lm>
1380 printbk$;                         <gp>
1390 open1,c,b,a$:printwt$;            <ml>
1400 gosub3200:ifxthen1660             <fp>
1410 sysu9,6,15,33,22,2:scnclr         <ck>
1420 a$=" loading... ":x=26:gosub2
430                                    <bc>
1430 print                             <na>
1440 ifc=1thensysu4,100                <kh>
1450 x=0                               <hc>
1460 fori=0to5                         <on>
1470 input#1,na$(i),fb$(i),rw(i),f
w(i),ma(i,0),ma(i,1),ma(i,2)           <eh>
1480 gosub3200:ifxthen1530             <fl>
1490 input#1,cp(i),ab(i),gd(i),sc(
i),lh(i),pk(i),ia(i),ik(i),ig(i)       <gn>
1500 gosub3200:ifxthen1530             <nb>
1510 input#1,vg(i),sk(i),we(i),fo(
i),e1(i),e2(i),e3(i),e4(i)             <ah>
1520 gosub3200                         <ha>
1530 ifxthen1660                       <bg>
1540 ifxtheni=5                        <he>
1550 next                              <kk>
1560 ifxthen1660                       <fm>
1570 input#1,rp,fp,kp,rw,fw,rw%,fw
%,na$,da,da$                           <df>
1580 gosub3200:ifxthen1660             <ej>
1590 input#1,gr,gf,f1,f2,f3,mo,ti$     <cl>
1600 gosub3200:ifxthen1660             <ko>
1610 close1                            <lb>
1620 gosub3200:ifxthen1660             <ei>
1630 scnclr:a$=" fertig ":x=26:gos
ub2430                                 <jp>
1640 sysu4,100                         <cc>
1650 goto3320                          <ii>
1660 close1                            <hm>
1670 sysu9,6,15,33,22,2:scnclr         <co>
1680 a$="ladefehler":x=26:gosub243
0                                      <gg>
1690 sysu4,200                         <ho>
1700 run                               <np>
1710 :                                 <kd>
1720 rem  neustart                     <lc>
1730 :                                 <ml>
1740 sysu9,22,19,27,21,1:print"nei
n"                                     <pa>
1750 sysu4,40                          <bk>
1760 sysu9,6,15,33,22,2:printcl$;      <id>
1770 a$="ihr name ?":x=26:gosub243
0                                      <ja>
1780 print:print:print"  >";:x=12:
gosub2270                              <lo>
1790 na$=a$                            <gn>
1800 sysu4,30                          <nf>
1810 printcl$;                         <ih>
1820 a$="firmenname ?":x=26:gosub2
430                                    <bb>
```

```
1830 print:print:print"  >";:x=10:
gosub2270                        <dp>
1840 c$=a$                       <kn>
1850 sysu4,30                    <ea>
1860 printcl$;                   <mb>
1870 a$="firmenkuertzel ?":x=26:go
sub2430                          <mc>
1880 print:print:print"  >";:x=3:g
osub2270:b$=a$                   <cn>
1890 sysu4,30                    <dk>
1900 scnclr                      <mo>
1910 a$="spieldauer ?":x=26:gosub2
430                              <ce>
1920 sysu9,12,19,14,21,0:printrv$"
-"                               <ip>
1930 sysu9,25,19,27,21,0:printrv$"
+"                               <fg>
1940 a$="2:00"                   <cp>
1950 sysu9,17,19,22,21,0:printrv$a
$                                <pf>
1960 sysu6                       <ng>
1970 sysu7,12,19,14,21,2010      <mm>
1980 sysu7,25,19,27,21,2070      <hg>
1990 sysu7,17,19,22,21,2120      <do>
2000 goto1960                    <im>
2010 ifa$="0:15"then1960         <fa>
2020 x=val(right$(a$,2))         <hc>
2030 x=x-15:ifx=-15thenx=45:a$=chr
$(asc(a$)-1)+":45":goto1950      <gb>
2040 a$=left$(a$,2)              <gi>
2050 ifxthena$=a$+right$(str$(x),2
):elsea$=a$+"00"                 <oc>
2060 goto1950                    <cd>
2070 ifa$="9:45"then1960         <po>
2080 x=val(right$(a$,2))         <do>
2090 x=x+15:ifx=60thenx=0:a$=chr$(
asc(a$)+1)+":00":goto1950        <hc>
2100 a$=left$(a$,2)+right$(str$(x)
,2)                              <ii>
2110 goto1950                    <fi>
2120 sysu9,17,19,22,21,1:printa$ <eo>
2130 sysu4,35                    <id>
2140 a$="0"+left$(a$,1)+right$(a$,
2)+"00"                          <mi>
2150 ti$=a$:da=ti:da$=a$         <ob>
2160 ti$="000000"                <ad>
2170 restore15950                <gm>
2180 fori=0to5                   <le>
2190 readna$(i),fb$(i),rw(i),fw(i)
,ma(i,0),ma(i,1),ma(i,2),cp(i),ab(
i),gd(i)                         <cb>
2200 readsc(i),lh(i),pk(i),ia(i),i
k(i),ig(i),vg(i),e1(i),e2(i),e3(i)
,e4(i)                           <lb>
2210 next                        <nh>
2220 na$(0)=c$:fb$(0)=right$("   "
+b$,3)                           <jd>
2230 goto3320                    <fl>
```

```
2240 :                           <ml>
2250 rem  input                  <mn>
2260 :                           <pd>
2270 a$=""                       <cp>
2280 getkeyb$                    <kc>
2290 ifinstr(zz$,b$)andlen(a$)<xth
enprintb$;:a$=a$+b$:goto2280     <fm>
2300 ifb$=de$andlen(a$)thenprintb$
;:a$=left$(a$,len(a$)-1):goto2280 <mc>
2310 ifb$=rt$thenreturn          <dh>
2320 goto2280                    <lg>
2330 :                           <ia>
2340 rem  datum                  <ma>
2350 :                           <ki>
2360 a$=str$(d)+c$:d=d+int(rnd(1)*
2+1.5)                           <ne>
2370 printa$                     <gd>
2380 print                       <ed>
2390 return                      <em>
2400 :                           <am>
2410 rem  ask                    <jo>
2420 :                           <de>
2430 printro$spc((x-len(a$))/2)lef
t$(li$,len(a$))                  <hc>
2440 printspc((x-len(a$))/2)rv$a$ <km>
2450 return                      <mf>
2460 :                           <if>
2470 rem  print                  <hl>
2480 :                           <kn>
2490 print" ";:vol6              <be>
2500 fori=1tolen(a$)             <jh>
2510 ifmid$(a$,i,1)<>" "thensysu8:
sound3,800,2                     <em>
2520 ifpeek(1277)then2520        <bk>
2530 printmid$(a$,i,1);          <ge>
2540 next                        <gn>
2550 print                       <ji>
2560 return                      <kc>
2570 :                           <gc>
2580 rem  ja-nein abfrage        <eg>
2590 :                           <ik>
2600 readx                       <cp>
2610 sysu5,0,1:printec$"m";      <bm>
2620 sysu9,6,16-x,33,22,2:scnclr <kj>
2630 sysu9,10,19,15,21,0:printrv$"
 ja "                            <jp>
2640 sysu9,24,19,29,21,0:printrv$"
nein"                            <an>
2650 sysu9,6,16-x,33,22,2        <pp>
2660 print" "left$(li$,24)       <ic>
2670 fori=1tox                   <ke>
2680 reada$                      <ni>
2690 print" "rv$a$               <df>
2700 next                        <kp>
2710 sysu6                       <fo>
2720 sysu7,10,19,15,21,2750 :rem
ja                               <jg>
2730 sysu7,24,19,29,21,2790 :rem
```

```
nein                                  <pn>
2740 goto2710                         <gb>
2750 sysu9,10,19,15,21,1:print" ja
 ":x=1                                <bj>
2760 sysu4,40                         <ke>
2770 sysu5,1,0                        <ga>
2780 return                           <fl>
2790 sysu9,24,19,29,21,1:print"nei
n":x=0                                <gm>
2800 goto2760                         <ok>
2810 :                                <ec>
2820 rem  balken                      <ch>
2830 :                                <gm>
2840 z=z+1:i=1                        <cd>
2850 ifz<8then2890                    <of>
2860 fori=1toint(z/8)                 <nl>
2870 char,x,y+1-i,rv$+"    "+ro$      <cl>
2880 next                             <bi>
2890 char,x,y+1-i,gr$(z-int(z/8)*8
)                                     <ee>
2900 return                           <em>
2910 readb:c=2*int((len(str$(y))-(
b>9))/2+.4)                           <ci>
2920 sysu5,0,1:printec$"m";           <kf>
2930 sysu9,6,16-b-(b>9)*10,33,22,2
:scnclr                               <kj>
2940 print" "left$(li$,24)            <ag>
2950 fori=1tob+(b>9)*10               <kl>
2960 reada$:print" "rv$a$             <kf>
2970 next                             <mn>
2980 a$=left$("########",c):ifb>9t
henmid$(a$,1,1)="$"                   <pm>
2990 sysu9,9,19,11,21,0:printrv$"-
"                                     <ne>
3000 sysu9,28,19,30,21,0:printrv$"
+"                                    <gk>
3010 sysu9,19-c/2,19,20+c/2,21,0      <fl>
3020 printusingrv$+a$;a               <hj>
3030 sysu6                            <ge>
3040 sysu7,9,19,11,21,3080            <dh>
3050 sysu7,28,19,30,21,3100           <go>
3060 sysu7,19-c/2,19,20+c/2,21,312
0                                     <jb>
3070 goto3030                         <kg>
3080 ifa-z>=xthena=a-z                <fa>
3090 goto3020                         <el>
3100 ifa+z<=ythena=a+z                <ao>
3110 goto3020                         <mf>
3120 sysu9,19-c/2,19,20+c/2,21,1      <ag>
3130 printusinga$;a                   <kg>
3140 sysu4,40                         <ah>
3150 sysu5,1,0                        <cm>
3160 return                           <fg>
3170 :                                <bg>
3180 rem  fehlerkontrolle             <oh>
3190 :                                <dp>
3200 x=0                              <fj>
3210 ifc=8thenifdsthenx=1             <ib>
3220 ifc=1thenifstthenx=1             <hi>
3230 return                           <od>
3240 :                                <kd>
3250 rem  warten                      <km>
3260 :                                <ml>
3270 ifjoy(2)=0then3270               <lb>
3280 return                           <eh>
3290 :                                <ah>
3300 rem  hauptprogramm               <cg>
3310 :                                <cp>
3320 sysu2                            <hh>
3330 mo=mo+1                          <pd>
3340 dg=0:db=0:dp=0:dl=0:dk=0:gw=0
:df=0                                 <ln>
3350 fori=0toaz                       <cc>
3360 gw(i)=-sc(i)*.05:dg=dg+gd(i)-
sc(i):gw=gw+we(i):df=df+fo(i)         <bk>
3370 db=db+ab(i):dp=dp+ma(i,0)+ma(
i,1)*2+ma(i,2)*3:dl=dl+lh(i):dk=dk
+pk(i)                                <gl>
3380 next                             <ae>
3390 ifam<dbthenam=db:al=0:goto343
0                                     <bb>
3400 al=am-db                         <ac>
3410 ifal<am/5thenam=am+1:al=al+1     <dn>
3420 ifam/al<4andal>2thenam=am-1:a
l=al-1                                <ji>
3430 x=az+1:dg=dg/x:db=db/x:dp=dp/
x:dl=dl/x:dk=dk/x:df=df/x             <kd>
3440 kp=kp-al*500                     <od>
3450 fori=0toaz                       <fe>
3460 ifsk(i)thenkp=kp-800*ab(i):el
sekp=kp+f3*ab(i)                      <lk>
3470 next                             <lj>
3480 ifkp<0thenfori=0toaz:sk(i)=0:
next:goto3850                         <hl>
3490 f3=100:ifkp>100000thenf3=80:i
fkp>250000thenf3=50                   <ko>
3500 c$=""                            <ln>
3510 fori=0toaz                       <dc>
3520 y=-1                             <pg>
3530 forx=0toaz                       <bo>
3540 ifinstr(c$,chr$(x))=0andlh(x)
>ytheny=lh(x):z=x                     <bj>
3550 next                             <fj>
3560 c$=c$+chr$(z)                    <ho>
3570 next                             <ib>
3580 fori=0toaz                       <bf>
3590 x(i)=asc(right$(c$,i+1))         <pb>
3600 next                             <lo>
3610 dd=ti/da:dm=int(mo/dd+.5):ift
i<500thendm=int(da/2000)              <fl>
3620 e=int(kp/800)-al*5               <hb>
3630 fori=0toaz                       <jd>
3640 z=x(i):ifkp<al*500orab(z)=0or
(lh(z)>=sk(z)andsk(z)<>0)thensk(z)
=0:goto3830                           <ha>
3650 c=ma(z,0)+ma(z,1)*2+ma(z,2)*3    <hl>
```

```
3660 a=c/ab(z)   :rem rationalisier
ung                                  <ao>
3670 b=c/dp      :rem produktionskr
aft                                  <gk>
3680 c=pk(z)/dk  :rem produktionsko
sten                                 <jo>
3690 d=(gd(z)-sc(z))/dg :rem finan
zen                                  <od>
3700 x=(d*3+c-b*2)/a:ifx<.3thenx=.
3                                    <pg>
3710 y=5+int(sqr(dm)+5)*x*(1.5-dd)  <hi>
3720 a=50*int(dl/50+1)              <di>
3730 ifa<lh(z)thensk(z)=0:goto3830  <jp>
3740 ifa<lh(z)+100thena=lh(z)+100   <hi>
3750 ifia(z)<.2thena=a+50           <kl>
3760 ifsk(z)then3810                <eg>
3770 ife-y*ab(z)<0then3830          <nc>
3780 e=e-y*ab(z)                    <bf>
3790 sk(z)=a                        <ae>
3800 goto3830                       <he>
3810 ife<0thensk(z)=0:goto3830      <ij>
3820 e=e-y*ab(z)+5                  <lp>
3830 next                           <ik>
3840 sk=0                           <ei>
3850 fori=0toaz                     <bn>
3860 ifsk(i)thenia(i)=ia(i)*.95:ig
(i)=1-(1-ig(i))*.98:sk=sk+1         <bk>
3870 iflh(i)>dlthenia(i)=1-(1-ia(i
))/(lh(i)/dl):elseia(i)=ia(i)*(lh(
i)/dl)                              <fo>
3880 iflh(i)>dlthenia(i)=ia(i)+(1-
ia(i))*(1-dl/lh(i)):elseia(i)=ia(i
)*lh(i)/dl                          <pn>
3890 next                           <ae>
3900 gr=gr+rw:ifrw>rw%thengr=gr-rw
+rw%                                <ed>
3910 x=gr/mo                        <hk>
3920 y=int(250*((rw+x)/rw%))        <lg>
3930 rp=int(rp+((y-rp)/3)+.5)       <gi>
3940 rw%=rw%+(rw%>rw)-(rw%<rw-2)    <ea>
3950 fw%=fw%+int(sqr(gw/1000))      <fn>
3960 gf=gf+fw:iffw>fw%thengf=gf-fw
+fw%                                <na>
3970 x=gf/mo                        <fi>
3980 y=int(10500*(fw%/(fw+fw+x)))   <ka>
3990 fp=int(fp+((y-fp)/3)+.5)       <ml>
4000 fw%=fw%+(fw%>fw)-(fw%<fw-3)    <nm>
4010 rw=rw%:fw=fw%                  <ip>
4020 :                              <mb>
4030 rem  meldungen                 <ij>
4040 :                              <oj>
4050 sysu9,0,0,39,24,2:printcl$;    <be>
4060 a$=" t e l e x ":x=38:gosub24
30                                  <nb>
4070 print:printec$"t"ec$"l"        <dd>
4080 sysu3                          <fe>
4090 a=int((mo-1)/12):b=mo-a*12:a=
a+1988                              <gb>
4100 a$=str$(a):mid$(a$,1,1)="."    <ji>
4110 b$=str$(b):mid$(b$,1,1)="."    <om>
4120 c$=b$+a$:d=1                   <co>
4130 ifsk(0)=0then4340              <lh>
4140 ifsk=az+1then4250              <aj>
4150 gosub2360                      <ik>
4160 a$=na$(0)+"-belegschaft im st
reik":gosub2490                     <ob>
4170 print                          <en>
4180 a$="die gewerkschaft verlangt
 eine":gosub2490                    <of>
4190 a$="anpassung der loehne von
derzeit":gosub2490                  <ch>
4200 a$="$"+str$(lh(0))+" auf $"+s
tr$(sk(0))+".":gosub2490            <ao>
4210 ifia(0)<.1thena$="mit gewalts
amen aktionen ist zu":gosub2490     <da>
4220 ifia(0)<.1thena$="rechen.":go
sub2490                             <hl>
4230 print:print                    <ci>
4240 goto4340                       <ld>
4250 gosub2360                      <ce>
4260 a$="generalstreik":gosub2490   <fl>
4270 print                          <bg>
4280 a$="dem aufruf der gewerkscha
ft, die":gosub2490                  <gf>
4290 a$="arbeit niederzulegen, fol
gten alle":gosub2490                <no>
4300 a$="arbeiter des landes.":gos
ub2490                              <bn>
4310 a$="von "+na$(0)+" fordert ma
n eine":gosub2490                   <bi>
4320 a$="lohnerhoehung auf $"+str$
(sk(0))+".":gosub2490               <lp>
4330 print:print:goto4440           <ki>
4340 ifsk+(sk(0)<>0)=0then4440      <jp>
4350 gosub2360                      <jf>
4360 a$="streiks in anderen untern
ehmen":gosub2490                    <ai>
4370 print                          <np>
4380 a$="gestreikt wird momentan b
ei:":gosub2490                      <fo>
4390 a$="":fori=1toaz               <mp>
4400 ifsk(i)thena$=a$+na$(i)+", "   <fp>
4410 next                           <bi>
4420 a$=left$(a$,len(a$)-2)+".":go
sub2490                             <oa>
4430 print:print                    <gk>
4440 iff1thenf1=f1-1:goto4610       <io>
4450 ifsk(0)=0oria(0)>.09then4610   <ff>
4460 ifrnd(1)*ia(0)>.05then4610     <hp>
4470 gosub2360                      <ma>
4480 a$="sachbeschaedigung":gosub2
490                                 <pk>
4490 print                          <na>
4500 a$="unbekannte taeter beschae
digten":gosub2490                   <ka>
4510 a$="waehrend einer streikdemo
```

```
nstration":gosub2490                    <hf>
4520 a$="eine maschine.":gosub2490      <eh>
4530 x=1000*int(rnd(0)*5+3)             <ko>
4540 a$="der schaden betraegt $"+s
tr$(x)+".":gosub2490                    <bh>
4550 print:print                        <co>
4560 gw(0)=gw(0)-x                      <pb>
4570 gosub3270                          <ed>
4580 restore15370:gosub2600             <il>
4590 ifx=0then4610                      <ka>
4600 f1=10:ia(0)=ia(0)*.6:ig(0)=ig
(0)+(1-ig(0))*.2                        <gn>
4610 ifrnd(0)*fo(0)*(pk(0)-200)<50
0000orpk(0)<300orrnd(1)<.7orsk(0)t
hen4720                                 <eb>
4620 gosub2360                          <jh>
4630 a$="erfolg der forschungsabte
ilung":gosub2490                        <ca>
4640 print                              <pn>
4650 a$="durch langwierige forschu
ng ist es":gosub2490                    <pc>
4660 a$="abteilung fuer planung, f
orschung":gosub2490                     <bk>
4670 a$="und entwicklung gelungen,
 die kosten":gosub2490                 <ai>
4680 a$="fuer die produnktion um $
 50 pro ware":gosub2490                <pd>
4690 a$="auf $"+str$(pk(0)-50)+" z
u senken.":gosub2490                    <ba>
4700 print:print                        <ob>
4710 pk(0)=pk(0)-50                     <df>
4720 ifsk(0)orrnd(1)>ab(0)/30orrnd
(0)<.8ormo<8then4860                    <hg>
4730 c=int(rnd(0)*3)                    <nl>
4740 ifma(0,c)=0then4860                <bh>
4750 ma(0,c)=ma(0,c)-1                  <jj>
4760 gosub2360                          <ce>
4770 a$="maschinenschaden":gosub24
90                                      <ok>
4780 print                              <bg>
4790 a$="eine maschine des typs "+
chr$(65+c)+", welche":gosub2490         <dh>
4800 a$=chr$(49+c)+" rohware"           <lc>
4810 ifcthena$=a$+"n"                   <fl>
4820 a$=a$+" erzeugt ist nicht meh
r":gosub2490                            <bp>
4830 a$="funktionsfaehig.":gosub24
90                                      <je>
4840 a$="eine reperatur ist nicht
rentabel.":gosub2490                    <ep>
4850 print:print                        <jd>
4860 ifrnd(0)<.98orrw(0)<5then4950      <lg>
4870 gosub2360                          <pg>
4880 a$="brand im rohwarenlager":g
osub2490                                <gn>
4890 print                              <pd>
4900 a$="heute nacht brannte das r
ohwarendepot":gosub2490                 <mk>
4910 a$="voellig aus.":gosub2490        <eo>
4920 a$="der schaden betraegt rund
 $"+str$(1000*int((rw(0)*rp)/1000+
.5))+".":gosub2490                      <mf>
4930 rw(0)=0                            <eg>
4940 print:print                        <gk>
4950 ifrnd(0)<.98orfw(0)<5then5150      <ef>
4960 gosub2360                          <cj>
4970 a$="brand im fertigwarenlager
":gosub2490                             <ld>
4980 print                              <ki>
4990 a$="von den flammen wurde heu
te frueh das":gosub2490                 <mf>
5000 a$="fertigwaren-magazin heimg
esucht.":gosub2490                      <gj>
5010 ifrnd(1)<.8then5040                <mg>
5020 a$="alle waren wurden vernich
tet.":gosub2490                         <oo>
5030 c=fw(0):goto5120                   <kn>
5040 c=int(rnd(0)*fw(0)+3):ifc+2>f
wthen5020                               <po>
5050 ifc<5then5090                      <mo>
5060 a$="von den"+str$(fw(0))+" wa
ren konnten"+str$(fw(0)-c):gosub24
90                                      <ik>
5070 a$="gerettet werden.":gosub24
90                                      <am>
5080 goto5120                           <ae>
5090 a$="die ware konnte gerettet
werden.":gosub2490                      <bp>
5100 print:print                        <gp>
5110 goto5150                           <lo>
5120 a$="der schaden betraegt rund
 $"+str$(1000*int((c*fp)/1000+.5))
+".":gosub2490                          <ep>
5130 print:print                        <ge>
5140 fw(0)=fw(0)-c                      <mc>
5150 c=rnd(1)*we(0)                     <jb>
5160 ifc<5000orrnd(0)<.8then5370        <bp>
5170 c=int(sqr(c/250)):b=100*int(f
p/100+2):ifc>20thenc=20                 <ni>
5180 gosub2360                          <mn>
5190 a$="auftrag":gosub2490             <pk>
5200 print                              <gb>
5210 a$="ein kunde erteilt einen a
uftrag":gosub2490                       <li>
5220 a$="ueber"+str(c)+" waren und
 bietet":gosub2490                      <ad>
5230 a$="$"+str$(b)+" bei sofortig
er lieferung.":gosub2490                <pg>
5240 iffw(0)>=cthen5310                 <fi>
5250 a$=na$(0)+" kann momentan nic
ht liefern.":gosub2490                  <ld>
5260 a$="der auftrag wurde an eine
 andere":gosub2490                      <pm>
5270 a$="firma vergeben.":gosub249
0                                       <nh>
5280 print:print                        <bf>
```

```
5290 ig(0)=ig(0)*.9:ik(0)=ik(0)*.7 <pd>
5300 goto5370                      <lc>
5310 print:print                   <ag>
5320 gosub3270                     <gf>
5330 restore15410:gosub2600        <ni>
5340 ifx=0then5290                 <aj>
5350 ik(0)=ik(0)+(1-ik(0))*.2      <pf>
5360 fw=fw-c:fw(0)=fw(0)-c:gw(0)=g
w(0)+b                             <bb>
5370 ifrnd(0)*(ig(0)↑2)<.25orrnd(1
)<.9orig(1)<.6ormo<3then5500       <hi>
5380 gosub2360:c=1000*int(rnd(0)*2
0+20)                              <gc>
5390 a$="sonderangebot":gosub2490  <eo>
5400 print                         <pd>
5410 a$="ein geschaeftsfreund biet
et ihnen":gosub2490                <hi>
5420 a$="eine maschine des typs b
um nur":gosub2490                  <jc>
5430 a$="$"+str$(c)+"an.":gosub249
0                                  <gf>
5440 print:print                   <bk>
5450 gosub3270                     <lf>
5460 restore15510:gosub2600        <pa>
5470 ifx=0then5500                 <ae>
5480 gw(0)=gw(0)-c:ma(0,0)=ma(0,0)
+1                                 <cb>
5490 ig(0)=ig(0)+(1-ig(0))*.3      <kp>
5500 ifrnd(1)<.95ormo<7orgd(0)<500
0then5580                          <mb>
5510 gosub2360:c=1000*int(rnd(1)*1
0+5)                               <mh>
5520 a$="steuernachzahlung":gosub2
490                                <pn>
5530 print                         <pi>
5540 a$="sie muessen $"+str$(c)+"
vermoegenssteuer":gosub2490        <ob>
5550 a$="nachzahlen.":gosub2490    <pk>
5560 gw(0)=gw(0)-c                 <ec>
5570 print:print                   <co>
5580 ifrnd(0)*sc(0)<30000orrnd(1)<
.95then5660                        <fh>
5590 gosub2360:c=1000*int(rnd(1)*5
+3)                                <im>
5600 a$="subventionen":gosub2490   <kc>
5610 print                         <ji>
5620 a$="der staat hat "+na$(0)+"
$"+str$(c):gosub2490               <li>
5630 a$="subventionen bewilligt.":
gosub2490                          <bc>
5640 gw(0)=gw(0)+c                 <dc>
5650 print:print                   <ld>
5660 iflh(0)<dl-50andrnd(1)<.3anda
b(0)>1then5680                     <ki>
5670 ifrnd(0)*(1-ia(0)*ia(0))<.1or
rnd(1)<.8oral>am/4orab(0)<3orsk(0)
then5750                           <pc>
5680 gosub2360                     <ik>
5690 a$="kuendigung":gosub2490     <li>
5700 print                         <en>
5710 a$="einer ihrer arbeiter kuen
digt seine":gosub2490              <pf>
5720 a$="stellung bei "+na$(0)+" f
ristlos.":gosub2490                <en>
5730 al=al+1:ab(0)=ab(0)-1         <nj>
5740 print:print                   <ih>
5750 ifal/am*ia(0)*rnd(1)<.02orlh(
0)<dl+50orrnd(1)<.8then5860        <ng>
5760 gosub2360                     <kg>
5770 a$="stellengesuch":gosub2490  <ck>
5780 print                         <oo>
5790 a$="ein bereits eingeschulter
 arbeiter":gosub2490               <na>
5800 a$="hat sich um eine anstellu
ng beworben.":gosub2490            <lk>
5810 print:print                   <lg>
5820 gosub3270                     <cd>
5830 restore15540:gosub2600        <lp>
5840 ifx=0thenia(0)=ia(0)*.98:goto
5860                               <ko>
5850 ab(0)=ab(0)+1:ia(0)=ia(0)+(1-
ia(0))*.05                         <dd>
5860 ifrnd(1)*ig(0)<.5orrnd(0)<.95
orrw(0)>50then6010                 <gn>
5870 c=int(rnd(1)*(ma+5)+10)       <mj>
5880 ifc+rw(0)>60thenc=60-rw(0)    <em>
5890 b=100*int(c*rp/110)           <ed>
5900 gosub2360                     <dg>
5910 a$="sonderangebot":gosub2490  <im>
5920 print                         <ah>
5930 a$="ein geschaeftsfreund biet
et ihnen"+str$(c):gosub2490        <dg>
5940 a$="rohwaren um $"+str$(b)+"
an.":gosub2490                     <mi>
5950 print:print                   <bk>
5960 gosub3270                     <lf>
5970 restore15570:gosub2600        <ph>
5980 ifx=0thenig(0)=ig(0)*.99:goto
6010                               <ln>
5990 rw(0)=rw(0)+c:gw(0)=gw(0)-b   <ij>
6000 ig(0)=ig(0)+(1-ig(0))*.95     <ka>
6010 ifpk(0)*rnd(0)*(ig(0)+1)<600o
rpk(0)<350orrnd(1)<.98then6150     <ll>
6020 gosub2360:c=1000*int(rnd(1)*4
+4000/(pk(0)-100)+2)               <ji>
6030 a$="angebot von forschungserg
ebnissen":gosub2490                <fn>
6040 print                         <pi>
6050 a$="ein forschungsinstitut bi
etet ihnen":gosub2490              <mf>
6060 a$="fuer $"+str$(c)+" die pla
ene fuer ein":gosub2490            <nf>
6070 a$="neues produktionsverfahre
n an, mit":gosub2490               <ph>
6080 a$="dem sie die produktionsko
sten um $ 50":gosub2490            <oh>
```

```
6090 a$="auf $"+str$(pk(0)-50)+" s
enken koennten.":gosub2490          <mg>
6100 print:print                    <mn>
6110 gosub3270                      <ho>
6120 restore15570:gosub2600         <hd>
6130 ifx=0then6150                  <op>
6140 gw(0)=gw(0)-c:pk(0)=pk(0)-50   <mb>
6150 ifrnd(1)*pk(0)<600orrnd(1)<.9
5ormo<5then6270                     <ma>
6160 gosub2360:c=500*int(rnd(0)*6+
ma+2)                               <ed>
6170 a$="umweltverschmutzung":gosu
b2490                               <ng>
6180 print                          <bb>
6190 a$="sie wurden wegen umweltve
rschmutzung":gosub2490              <db>
6200 a$="angezeigt und fuer schuld
ig befunden.":gosub2490             <cf>
6210 a$="sie wurden zu einer gelds
trafe von":gosub2490                <ok>
6220 a$="$"+str$(c)+" verurteilt."
:gosub2490                          <ca>
6230 a$="der grund fuer die versch
mutzung lag":gosub2490              <fb>
6240 a$="im veralterten produktion
sverfahren.":gosub2490              <fc>
6250 print:print                    <ih>
6260 gw(0)=gw(0)-c                  <hj>
6270 f2=0:ifrnd(0)<.98orma<5then63
80                                  <na>
6280 gosub2360:c=1000*int(rnd(1)*5
+ma/2+2)                            <jb>
6290 a$="erpressung":gosub2490      <kn>
6300 print                          <ac>
6310 a$="ein unbekannter fordert $
"+str$(c)+" und":gosub2490          <dm>
6320 a$="und droht bei nichterfuel
lung seiner":gosub2490              <np>
6330 a$="forderungen mit einem bom
benanschlag.":gosub2490             <ji>
6340 print:print                    <fg>
6350 gosub3270                      <kc>
6360 restore15600:gosub2600         <pl>
6370 f2=x+1                         <pn>
6380 iff2=0or(f2=1andrnd(0)<.6)or(
f2=2andrnd(0)<.95)then6520          <ja>
6390 gosub2360:c=int(ma/3-rnd(1)):
ifc<2thenc=2                        <lk>
6400 a$="bombenanschlag":gosub2490  <nf>
6410 print                          <np>
6420 fori=1toc                      <og>
6430 x=int(rnd(1)*2.5)              <pg>
6440 ifma(0,x)>0thenma(0,x)=ma(0,x
)-1:else6430                        <bi>
6450 next                           <bi>
6460 a$="heute nacht explodierte i
n der":gosub2490                    <ij>
6470 a$="maschinenhalle ein spreng
satz, wobei":gosub2490              <ak>
6480 a$=right$(str$(c),1)+" maschi
nen zerstoert wurden.":gosub2490    <nk>
6490 a$="personen kamen nicht zu s
chaden.":gosub2490                  <pa>
6500 iff2=2thena$="der erpresser h
at sein wort nicht":gosub2490:a$="
gehalten.":gosub2490                <ka>
6510 print:print                    <la>
6520 ifrnd(0)<.98orgd(0)<100000the
n6620                               <ce>
6530 gosub2360:c=1000*int(gd(0)/50
00*(rnd(1)+.2)):ifc>50000thenc=500
00                                  <ac>
6540 a$="diebstahl":gosub2490       <df>
6550 print                          <pi>
6560 a$="heute nacht wurde bei "+n
a$(0):gosub2490                     <ao>
6570 a$="eingebrochen.":gosub2490   <cc>
6580 a$="alles geld im tresor ($"+
str$(c)+") wurde":gosub2490         <na>
6590 a$="gestohlen":gosub2490       <jl>
6600 print:print                    <ho>
6610 gd(0)=gd(0)-c                  <ic>
6620 ifrnd(1)<.95orpk(0)>950then67
00                                  <ap>
6630 gosub2360:c=1000*int(rnd(1)*(
1200-pk(0))/100+1)                  <oa>
6640 a$="verkauf von know-how":gos
ub2490                              <fl>
6650 print                          <lp>
6660 a$="der verkauf von forschung
sergebnissen":gosub2490             <hc>
6670 a$="bracht ihnen $"+str$(c)+"
.":gosub2490                        <in>
6680 print:print                    <ad>
6690 gw(0)=gw(0)+c                  <hf>
6700 ifrnd(0)*pk(0)<400orrnd(1)<.9
7ormo<10then6790                    <cj>
6710 gosub2360                      <mn>
6720 a$="strompreiserhoehung":gosu
b2490                               <nf>
6730 print                          <gb>
6740 a$="durch die verteuerung des
 stromes":gosub2490                 <me>
6750 a$="erhoehen sich ihre produk
tionskosten":gosub2490              <pf>
6760 a$="um $ 50.":gosub2490        <cp>
6770 pk(0)=pk(0)+50                 <cd>
6780 goto6870                       <jk>
6790 ifrnd(0)*pk(0)<500orrnd(1)<.9
7then6870                           <al>
6800 gosub2360                      <ce>
6810 a$="strompreissenkung":gosub2
490                                 <pj>
6820 print                          <bg>
6830 a$="durch die senkung koennen
 sie ihre":gosub2490                <hb>
```

```
6840 a$="produktionskosten um $ 50
 senken.":gosub2490                  <kd>
6850 print:print                     <fg>
6860 pk(0)=pk(0)-50                  <fp>
6870 ifjoy(2)and128then7000          <cc>
6880 gosub2360                       <cc>
6890 a$="boersenbericht":gosub2490   <da>
6900 print                           <lf>
6910 a$="preis fuer rohwaren     :
$"+str$(rp):gosub2490                <ob>
6920 a$="preis fuer fertigwaren :
$"+str$(fp):gosub2490                <fj>
6930 ifjoy(2)and128then7000          <ff>
6940 print:print                     <cj>
6950 gosub2360                       <ol>
6960 a$="arbeitsmarkt":gosub2490     <nh>
6970 print                           <ed>
6980 a$="arbeitslose    :"+str$(al)
:gosub2490                           <pg>
6990 a$="beschaeftigte :"+str$(am-
al):gosub2490                        <db>
7000 print:print:print               <db>
7010 a$="ende der uebertragung":go
sub2490                              <ib>
7020 :                               <ej>
7030 rem  eingabemenues              <ij>
7040 :                               <hb>
7050 gosub3270                       <im>
7060 ma=ma(0,0)+ma(0,1)+ma(0,2)      <hk>
7070 pk=ma+ma(0,1)+ma(0,2)*2         <oo>
7080 sysu9,0,0,39,24,2:scnclr:prin
tec$"m";                             <kn>
7090 a$=" s t a t u s ":x=38:gosub
2430                                 <pm>
7100 print:print                     <co>
7110 printmo;le$". spielrunde:"      <ll>
7120 print" "left$(ul$,13+len(str$
(mo)))                               <ok>
7130 x=int(ti/3600+.5)               <kb>
7140 a=int(x/60):b=x-a*60            <pp>
7150 print" spielzeit ...........
:"a;le$":";:pudef"0":printusing"##
";b:pudef" "                         <lk>
7160 print" kapital .............
:"gd(0)+gw(0)"$"                     <hk>
7170 print" schulden ............
:"sc(0)"$"                           <bm>
7180 print" maschinen ...........
:"ma                                 <he>
7190 print" computer ............
:"cp(0)                              <db>
7200 print" arbeiter ............
:"ab(0)                              <be>
7210 print" rowwaren ............
:"rw(0)                              <lc>
7220 print" fertigwaren .........
:"fw(0)                              <lk>
7230 print" produktionakapazitaet
:"pk                                 <fg>
7240 print" lohnhoehe ...........
:"lh(0)"$"                           <hp>
7250 print" roduktionskosten ....
:"pk(0)"$"                           <pi>
7260 print" exclusivvertrieb ....
:"vg(0)"ware";:ifvg(0)<>1thenprint
"n":elseprint                        <jo>
7270 gosub3270                       <cj>
7280 restore15440:gosub2600          <lp>
7290 ifx=0then8910                   <pg>
7300 sysu9,0,0,39,24,2:scnclr        <kn>
7310 a$=" s t u d i e ":x=38:gosub
2430                                 <eo>
7320 print:print                     <bf>
7330 print" 1.)  beliebtheit des u
nternehmens"                         <lm>
7340 print"       kosten: $ 2000"    <ci>
7350 print                           <do>
7360 print" 2.)  vergleich mit der
 konkurrenz"                         <eg>
7370 print"       (produktionskoste
n, loehnen und"                      <he>
7380 print"       rationalisierung)
"                                    <pm>
7390 print"       kosten: $ 4000"    <eo>
7400 print                           <kb>
7410 print" 3.)  durchschnittliche
 werbe- und"                         <ol>
7420 print"       forschungsausgabe
n"                                   <mn>
7430 print"       kosten: $ 1000"    <fk>
7440 print                           <pd>
7450 print" 4.)  ausstattung der k
onkurrenz"                           <pm>
7460 print"       kosten: $ 5000"    <dn>
7470 fori=0to3                       <ea>
7480 sysu9,3+i*7,20,8+i*7,22,0:pri
ntrv$"nr."+chr$(49+i)                <hn>
7490 next                            <ea>
7500 sysu9,31,20,36,22,0:printrv$"
ende"                                <ea>
7510 sysu6                           <pc>
7520 e=0                             <ja>
7530 sysu7,3+e*7,20,8+e*7,22,7550    <ef>
7540 e=e+1:ife>4then7510:else7530    <jn>
7550 sysu9,3+e*7,20,8+e*7,22,1       <mg>
7560 ife=4thenprint"ende":sysu4,40
:goto8910:elseprint"nr."+chr$(49+e
)                                    <bg>
7570 one+1gosub7610,7830,8590,8690
,                                    <hc>
7580 sysu9,3+e*7,20,8+e*7,22,0:pri
ntrv$"nr."+chr$(49+e)                <go>
7590 goto7510                        <cb>
7600 :                               <nf>
7610 sysu5,0,1                       <om>
7620 sysu9,7,4,32,20,2:scnclr        <ha>
```

```
7630 a$="      beliebtheit        ":x
=24:gosub2430                          <fk>
7640 print" "left$(li$,22)             <oc>
7650 print" "rv$"gut      "ro$chr$(
186)                                   <nk>
7660 print                             <kl>
7670 print" "left$(li$,22)             <lm>
7680 print" "rv$"mittel  "ro$chr$(
186)                                   <fo>
7690 print                             <oj>
7700 print" "left$(li$,22)             <mk>
7710 print" "rv$"schlecht"ro$chr$(
186)" (a) (b) (c)"                     <ij>
7720 print                             <cf>
7730 print" a..bei belegschaft"        <je>
7740 print" b..bei konsumenten"        <jb>
7750 print" c..bei anderen firmen"     <nd>
7760 x=19:y=13:z=int(ia(0)*48):gos
ub2840                                 <ij>
7770 x=23:z=int(ik(0)*48):gosub284
0                                      <lo>
7780 x=27:z=int(ig(0)*48):gosub284
0                                      <ai>
7790 gosub3270                         <gl>
7800 sysu5,1,0                         <ip>
7810 gw(0)=gw(0)-2000                  <nb>
7820 return                            <od>
7830 sysu5,0,1                         <gb>
7840 sysu9,3,5,36,20,2:scnclr          <ia>
7850 a$="      produktionskosten
      ":x=32:gosub2430                 <cj>
7860 a$=" "+left$(li$,30)              <ei>
7870 printa$:print" "rv$"1000"ro$c
hr$(186)                               <hi>
7880 printa$:print" "rv$" 750"ro$c
hr$(186)                               <ic>
7890 printa$:print" "rv$" 500"ro$c
hr$(186)                               <ai>
7900 printa$:print" "rv$" 250"ro$c
hr$(186)                               <eb>
7910 printa$:print" "rv$"   0"ro$c
hr$(186)                               <im>
7920 printspc(6);                      <oo>
7930 fori=0toaz                        <jn>
7940 printusing" ###";pk(i)/(cp(i)
+1);                                   <jk>
7950 next                              <nm>
7960 print:printspc(6);                <cj>
7970 fori=0toaz                        <pb>
7980 print" "fb$(i);                   <jn>
7990 next                              <cm>
8000 y=16                              <bl>
8010 fora=0toaz                        <ae>
8020 x=11+a*4:z=int(pk(a)*.064/(cp
(a)+1))                                <bh>
8030 gosub2840                         <id>
8040 next                              <jb>
8050 gosub3270                         <al>
8060 sysu5,1,0                         <ab>
8070 sysu4,20                          <gj>
8080 sysu9,0,5,39,20,2:scnclr          <co>
8090 a$="               lohnhoehe
      ":x=38:gosub2430                 <np>
8100 a$=" "+left$(li$,36)              <nm>
8110 a=0:b=10000                       <mp>
8120 fori=0toaz                        <dm>
8130 iflh(i)>athena=lh(i)              <fp>
8140 iflh(i)<bthenb=lh(i)              <pe>
8150 next                              <gn>
8160 a=100*int(a/100+1)                <jp>
8170 c=100*int((a-b)/400+1)            <ba>
8180 b=a-c*4                           <fc>
8190 fori=0to4                         <nh>
8200 printa$:printusing" "+rv$+"##
##"+ro$+chr$(186);a-c*i                <kc>
8210 next                              <og>
8220 printspc(6);                      <km>
8230 fori=0toaz                        <go>
8240 printusing" ####";lh(i);          <ji>
8250 next                              <dg>
8260 print:printspc(6);                <me>
8270 fori=0toaz                        <hk>
8280 print"  "fb$(i);                  <db>
8290 next                              <ih>
8300 y=16                              <pc>
8310 forc=0toaz                        <ch>
8320 x=9+c*5                           <da>
8330 z=int((lh(c)-b)/(a-b)*64+.5)      <mk>
8340 gosub2840                         <ha>
8350 next                              <pp>
8360 gosub3270                         <no>
8370 sysu5,1,0                         <hf>
8380 sysu4,20                          <bg>
8390 sysu9,4,6,35,19,2:scnclr          <mh>
8400 a$="  fertigwaren pro arbeite
r  ":x=30:gosub2430                    <ig>
8410 a$=" "+left$(li$,28)              <ki>
8420 printa$:print" "rv$" 3"ro$chr
$(186)                                 <ji>
8430 print:printa$:print" "rv$" 2"
ro$chr$(186)                           <hm>
8440 print:printa$:print" "rv$" 1"
ro$chr$(186)                           <bj>
8450 printspc(4);                      <ih>
8460 fori=0toaz                        <bl>
8470 print" "fb$(i);                   <ep>
8480 next                              <ae>
8490 y=15                              <nn>
8500 fora=0toaz                        <ak>
8510 x=10+a*4:ifma(a,0)+ma(a,1)+ma
(a,2)=0then8540                        <md>
8520 z=int((ma(a,0)+ma(a,1)*2+ma(a
,2)*3)/(ma(a,0)+ma(a,1)+ma(a,2))*2
4-23.5)                                <ml>
8530 gosub2840                         <di>
8540 next                              <hm>
```

```
8550 ifjoy(2)=0then8550                 <df>
8560 sysu5,1,0                          <bi>
8570 gw(0)=gw(0)-4000                   <lf>
8580 return                             <nj>
8590 sysu5,0,1                          <ai>
8600 sysu9,8,9,30,16,2:scnclr           <lh>
8610 a$=" mittlere ausgaben ":x=21
:gosub2430                              <nj>
8620 print                              <de>
8630 printusing" werbung   : $ ####
#";gw/(az+1)                            <dp>
8640 printusing" forschung: $ ####
#";df                                   <oa>
8650 gosub3270                          <ed>
8660 sysu5,1,0                          <po>
8670 gw(0)=gw(0)-1000                   <gj>
8680 return                             <ka>
8690 sysu5,0,1                          <nc>
8700 sysu9,7,7,32,19,2                  <mf>
8710 fori=1toaz                         <kn>
8720 scnclr                             <em>
8730 a$=" "+na$(i)+" ":x=24:gosub2
430                                     <mi>
8740 print                              <cf>
8750 print" finanzen    :"gd(i)+gw(
i)-sc(i)"$"                             <ed>
8760 print" produktion  :"ma(i,0)+m
a(i,1)*2+ma(i,2)*3                      <af>
8770 print" computer    :"cp(i)         <pp>
8780 print" arbeiter    :"ab(i)         <nf>
8790 print" vertrieb    :"vg(i)"stu
eck"                                    <cl>
8800 print" rohwaren    :"rw(i)         <aa>
8810 print" fertigwaren:"fw(i)          <nj>
8820 sysu4,40                           <lb>
8830 gosub3270                          <on>
8840 next                               <nh>
8850 sysu5,1,0                          <ge>
8860 gw(0)=gw(0)-5000                   <mp>
8870 return                             <bp>
8880 :                                  <np>
8890 rem  produktion                    <jp>
8900 :                                  <ah>
8910 sysu9,0,0,39,24,2:scnclr           <ja>
8920 a$=" p r o d u k t i o n ":x=
38:gosub2430                            <kj>
8930 print:print                        <jd>
8940 sysu9,2,4,37,18,2                  <nm>
8950 c=0:ifsk(0)=0thengw(0)=gw(0)-
ab(0)*lh(0)                             <il>
8960 ifcp(0)thenprint" ";cp$:c=1        <eb>
8970 ifma=0then9060                     <li>
8980 fori=2to0step-1                    <kl>
8990 ifma(0,i)=0then9050                <cg>
9000 forz=1toma(0,i)                    <fg>
9010 y=int(c/7):x=c-y*7                 <ic>
9020 char,x*5+5,y*3+5,ma$(i)            <db>
9030 c=c+1                              <hi>
9040 next                               <gi>
9050 next                               <hm>
9060 ifab(0)=0then9110                  <pl>
9070 fori=0+cp(0)toab(0)-1+cp(0)        <pa>
9080 y=int(i/7):x=i-y*7                 <kf>
9090 char,x*5+3,y*3+5,ab$               <pf>
9100 next                               <ob>
9110 ifrw(0)<15theni=17:goto9150        <ja>
9120 fori=17to18-int(rw(0)/15)step
-1                                      <dc>
9130 char,3,i,rp$                       <dp>
9140 next                               <db>
9150 char,3,i,left$(rp$,rw(0)-(15*
(17-i)))                                <ia>
9160 iffw(0)<15theni=17:goto9200        <lm>
9170 fori=17to18-int(fw(0)/15)step
-1                                      <po>
9180 char,22,i,fp$                      <gi>
9190 next                               <je>
9200 char,22,i,left$(fp$,fw(0)-(15
*(17-i)))                               <on>
9210 char,0,20,""                       <fn>
9220 ifab(0)=0thena$="sie haben ke
ine arbeiter.":goto9430                 <ij>
9230 ifma=0thena$="sie haben keine
 maschinen.":goto9430                  <nd>
9240 ifsk(0)thena$="die arbeiter s
treiken.":goto9430                      <gg>
9250 ifrw(0)=0thena$="sie haben ke
ine rohware.":goto9430                  <aj>
9260 iffw(0)=60thena$="das fertigw
arenlager ist voll.":goto9430           <jl>
9270 a=0:y=0                            <hj>
9280 fori=2to0step-1                    <hj>
9290 ifma(0,i)=0then9330                <la>
9300 forx=1toma(0,i)                    <dj>
9310 a=a+1:ifa<=ab(0)theny=y+i+1        <mj>
9320 next                               <jl>
9330 next                               <kp>
9340 ifrw(0)<ytheny=rw(0)               <ah>
9350 if60-fw(0)<ytheny=60-fw(0)         <ie>
9360 printspc(4)"die maximale prod
uktionskapazitaet"                      <dm>
9370 printspc(11)"liegt bei"y"stue
ck."                                    <dk>
9380 gosub3270                          <no>
9390 restore15650:x=0:z=1:a=y:gosu
b2910                                   <hm>
9400 gw(0)=gw(0)-a*pk(0)/(cp(0)+1)      <kh>
9410 rw(0)=rw(0)-a:z(0)=a               <jf>
9420 goto9480                           <nj>
9430 gw(0)=gw(0)-ma*500:z(0)=0          <mi>
9440 printspc(2)"sie koennen leide
r nicht produzieren."                   <mg>
9450 print:printspc((40-len(a$))/2
)a$                                     <bi>
9460 :                                  <gk>
9470 gosub3270                          <bn>
```

```
9480 sysu2                                          <lo>
9490 fori=1toaz                                     <be>
9500 a=ma(i,0)+ma(i,1)+ma(i,2)                      <df>
9510 ifsk(i)thengw(i)=gw(i)-a*500:z(i)=0:goto9640   <mi>
9520 gw(i)=gw(i)-ab(i)*lh(i)                        <bc>
9530 ifa=0thenz(i)=0:goto9640                       <nn>
9540 a=0:b=0                                        <ii>
9550 forx=2to0step-1                                <mm>
9560 ifma(i,x)=0then9600                            <bj>
9570 fory=1toma(i,x)                                <bk>
9580 ifa<ab(i)thena=a+1:b=b+x                       <lo>
9590 next                                           <lj>
9600 next                                           <mn>
9610 ifb>rw(i)thenb=rw(i)                           <mj>
9620 ifb>60-fw(i)thenb=60-fw(i)                     <hd>
9630 z(i)=b:gw(i)=gw(i)-b*(pk(i)/(cp(i)+1)):rw(i)=rw(i)-b   <af>
9640 next                                           <bn>
9650 :                                              <oj>
9660 rem  ein- und verkauf                          <ao>
9670 :                                              <bb>
9680 a=10000*(fw%/(fw+x+50))                        <cd>
9690 b=250*((rw+(gr/mo))/rw%)                       <mm>
9700 fori=1toaz                                     <bn>
9710 c=ma(i,0)+ma(i,1)*2+ma(i,2)*3                  <hj>
9720 d=fp+(a-fp)/(3-e4(i)/8)                        <nb>
9730 e=rp+(b-rp)/(3-e4(i)/5)                        <hh>
9740 x=d-pk(i)/(cp(i)+1)-lh(i)*ab(i)/c              <km>
9750 ifx=<0thenx(i)=0:goto9900                      <gb>
9760 y=c*int(e4(i)/3+1)-rw(i)                       <ff>
9770 ifsk(i)theny=int((rw(i)+y)/2-rw(i)):ify<=0thenx(i)=0:goto9900   <kj>
9780 z=(e/rp)                                       <id>
9790 y=int(y+c*(1-z)+.5)                            <lp>
9800 z=z+vg(i)/5                                    <cj>
9810 z=z+(c+y+rw(i))/(rw(i)+c)-1                    <po>
9820 ifsk(i)thenz=1+(z-1)/2                         <jl>
9830 x=x-x/(z↑(e1(i)/5+1.5))                        <jb>
9840 ifx<=rpthenx(i)=0:goto9900                     <ef>
9850 ify+rw(i)>59theny=60-rw(i)                     <de>
9860 ify<0theny=0                                   <cp>
9870 x=x-(x-rp)/3:ifx>3*rpthenx=3*rp                <dj>
9880 ifx-rp>200thenx=x-(x-rp-200)*.6                <jo>
9890 x(i)=y:y(i)=int(x)                             <ap>
9900 next                                           <ch>
9910 sysu9,0,0,39,24,2:scnclr                       <li>
9920 a$=" e i n k a u f ":x=38:gosub2430            <oc>
9930 print:print                                    <pf>
9940 print" der offizielle marktpreis fuer eine"    <od>
9950 print" rohware betraegt $"rp;le$"."            <kh>
9960 print" es werden"rw%"rohwaren angeboten."      <bj>
9970 print" sie besitzen momentan"rw(0)"rohwaren."  <hc>
9980 print" sie koennen also ";                     <gc>
9990 ifrw(0)>59thenprint"keine rohwaren mehr":print" unterbringen":x(0)=0:goto10100   <og>
10000 print"maximal";:if60-rw(0)>rwthenprintrw;:elseprint60-rw(0);   <ea>
10010 print"rohware";:if60-rw(0)>1thenprint"n":elseprint   <ag>
10020 print" kaufen."                               <ed>
10030 sysu3                                         <po>
10040 restore15680:a=0:y=60-rw(0):ifrw<ytheny=rw    <le>
10050 x=0:z=1:gosub2910                             <pn>
10060 x(0)=a:ifa=0then10100                         <hd>
10070 restore15710:a=10*int(rp/10+.95)              <pp>
10080 x=a:y=9990:z=10:gosub2910                     <lc>
10090 y(0)=a                                        <mk>
10100 sysu5,0,1                                     <po>
10110 sysu9,5,6,34,19,2:scnclr                      <fn>
10120 a$="versteigerung":x=28:gosub2430             <fe>
10130 print:print" unternehmen  waren  preis"       <fl>
10140 print:x=0                                     <me>
10150 fori=0toaz                                    <el>
10160 x=x+x(i):ifx(i)=0theny(i)=0                   <bf>
10170 next                                          <ef>
10180 c=x                                           <cf>
10190 ifx<=rwtheny=rp:goto10270                     <no>
10200 z=10000                                       <gi>
10210 fori=0toaz                                    <bf>
10220 ify(i)<zandx(i)<>0thenz=y(i):b=i              <dn>
10230 next                                          <lm>
10240 ifx-x(b)>=rwthenx=x-x(b):x(b)=0:elsex(b)=x(b)-x+rw:x=rw   <mm>
10250 ifx>rwthen10200                               <nm>
10260 y=y(b)                                        <gg>
10270 fori=0toaz                                    <go>
10280 print" "na$(i)tab(19)x(i)tab(26)"$"le$;       <cc>
10290 ify(i)=0thenprint" ---":goto10310             <nn>
10300 ify(i)>=ythenprinty:elseprinty(i)             <ab>
10310 next                                          <fo>
10320 rw=c+rw%-rw                                   <le>
10330 ifx(0)=1thena$="":elsea$="n"                  <lj>
10340 char,2,21,"ihnen wurde"+a$+str$(x(0))+" ware"+a$+" zugeschlagen."   <im>
10350 char,2,22,"sie bezahlen $"+str$(y*x(0))+"."   <ce>
10360 fori=0toaz                                    <ob>
```

```
10370 gw(i)=gw(i)-y*x(i):rw(i)=rw(
i)+x(i)                            <nl>
10380 next                         <ol>
10390 gosub3270                    <kb>
10400 sysu5,1,0                    <hg>
10410 sysu4,40                     <ke>
10420 :                            <pd>
10430 rem  verkauf                 <in>
10440 :                            <bl>
10450 sysu2                        <ck>
10460 fori=0toaz                   <ip>
10470 ifvg(i)=0then10520           <ke>
10480 iffw(i)<vgthen10510          <ge>
10490 gw(i)=gw(i)+vg(i)*fp:fw(i)=f
w(i)-vg(i):fw=fw-vg(i):ik(i)=ik(i)
+(1-ik(i))*.05                     <dh>
10500 goto10520                    <ha>
10510 gw(i)=gw(i)+fw(i)*fp:fw=fw-f
w(i):fw(i)=0:ik(i)=ik(i)*.9        <fo>
10520 next                         <ae>
10530 iffw<0thenfw=0:fw%=fw%+1     <fn>
10540 a=10000*(fw%/(fw+x+50))      <pd>
10550 z=0:fori=0toaz:z=z+fw(i):nex
t                                  <fp>
10560 fori=1toaz                   <an>
10570 c=ma(i,0)+ma(i,1)*2+ma(i,2)*
3                                  <pi>
10580 b=fp+(a-fp)/(3-e4(i)/8)      <ke>
10590 d=y+pk(i)+lh(i)/c            <cp>
10600 iffp-d<20thenx(i)=0:goto1077
0                                  <al>
10610 e=b/a*3                      <hg>
10620 ifgd(i)-sc(i)+gw(i)<0thene=e
/2                                 <af>
10630 e=e+fw(i)/20:e=(3-e1(i)/10)*
e*dp/c                             <af>
10640 if60-fw(i)<candsk(i)=0thene=
e/3                                <cp>
10650 ifa-fw(i)<fw-4then10690      <ma>
10660 x(i)=fw(i)-fw(i)/e           <hp>
10670 x(i)=(x(i)*(b/a)*3)/3        <ai>
10680 goto10710                    <in>
10690 x(i)=fw(i):y(i)=fp           <md>
10700 goto10770                    <im>
10710 x(i)=int(x(i)):ifx(i)>fw(i)t
henx(i)=fw(i)                      <bi>
10720 ifx(i)<0thenx(i)=0           <hg>
10730 e=sqr(e):ife<1.3thene=1.3    <al>
10740 y(i)=fp-(fp-d)/e             <fb>
10750 y(i)=int(y(i)+120*we(i)/(gw+
1)*(az+1)+50*ik(i)):ify(i)>fptheny
(i)=fp                             <dj>
10760 iffp-y(i)>250theny(i)=int(y(
i)+(fp-y(i)-250)*.6)               <hg>
10770 next                         <pk>
10780 sysu9,0,0,39,24,2:scnclr     <op>
10790 a$=" v e r k a u f ":x=38:go
sub2430                            <nk>
10800 print:print                  <ec>
10810 print" der offizielle preis
fuer eine"                         <cf>
10820 print" fertigware betraegt $
"fp;le$"."                         <kd>
10830 print" auftraege ueber"fw"wa
ren wurden an"                     <il>
10840 print" die billigstbietenden
 vergeben."                        <ck>
10850 print" sie besitzen"fw(0)"fe
rtigware";                         <ce>
10860 iffw(0)=1thenprint".":elsepr
int"n."                            <mh>
10870 sysu3                        <fl>
10880 iffw(0)>0then10900           <im>
10890 gosub3270:x(0)=0:goto10990   <bl>
10900 restore15740                 <fd>
10910 z=1:x=0:y=fw:ify>fw(0)theny=
fw(0)                              <ao>
10920 a=y:gosub2910                <ok>
10930 ifa=0thenx(0)=0:goto10990    <no>
10940 x(0)=a                       <oj>
10950 restore15770                 <nf>
10960 a=10*int(fp/10):x=0:y=a:z=10 <oj>
10970 gosub2910                    <ga>
10980 y(0)=a                       <ol>
10990 sysu5,0,1                    <el>
11000 sysu9,5,6,34,19,2:scnclr     <bp>
                                   <gn>
11010 a$="versteigerung":x=28:gosu
b2430                              <gn>
11020 print:print" unternehmen  wa
ren  preis"                        <oc>
11030 print                        <bl>
11040 x=0                          <ep>
11050 fori=0toaz                   <ah>
11060 x=x+x(i):ifx(i)=0theny(i)=0  <kk>
11070 next                         <fe>
11080 c=x                          <ga>
11090 ifx<=fwtheny=fp:goto11170    <jf>
11100 z=0                          <dj>
11110 fori=0toaz                   <gi>
11120 ify(i)-(100*we(i)/(gw+1)*(az
+1)+200*ik(i))>zandx(i)<>0thenz=y(
i):b=i                             <kh>
11130 next                         <mn>
11140 ifx-x(b)>=fwthenx=x-x(b):x(b
)=0:elsex(b)=x(b)-x+fw:x=fw        <ih>
11150 ifx>fwthen11100              <nk>
11160 y=y(b)                       <ei>
11170 fori=0toaz                   <el>
11180 print" "na$(i)tab(19)x(i)tab
(26)"$"le$;                        <nk>
11190 ify(i)=0thenprint" ----":got
o11210                             <jd>
11200 ify(i)<ythenprinty:elseprint
y(i)                               <jj>
11210 next                         <gn>
```

```
11220 fw=c+fw%-fw                        <pc>
11230 ifx(0)=1thena$="":elsea$="n"       <om>
11240 char,2,21,"sie koennen"+str$
(x(0))+" ware"+a$+" verkaufen."          <lk>
11250 char,2,22,"sie nehmen $"+str
$(x(0)*(-y(0)*(y(0)>y)-y*(y>=y(0))
))+" ein."                               <be>
11260 fori=0toaz                         <ce>
11270 fw(i)=fw(i)-x(i)                   <pf>
11280 ify(i)<ythengw(i)=gw(i)+x(i)
*y:elsegw(i)=gw(i)+x(i)*y(i)             <ad>
11290 ify(i)>0andx(i)=0thenik(i)=i
k(i)*.97                                 <oo>
11300 ify(i)>0andx(i)>0thenik(i)=i
k(i)+(1-ik(i))*.02                       <jk>
11310 next                               <dg>
11320 gosub3270                          <fe>
11330 sysu5,1,0                          <ck>
11340 sysu4,40                           <aj>
11350 :                                  <dp>
11360 rem  investitionen                 <db>
11370 :                                  <gh>
11380 sysu9,0,0,39,24,2:scnclr           <kn>
11390 a$=" sachwerte und personal
":x=38:gosub2430                         <mp>
11400 print:print                        <bf>
11410 print" "cp$"  computer
    $ 150000"                            <db>
11420 printspc(5)"(halbe produktio
nskosten)"                               <ef>
11430 print                              <do>
11440 print" "ma$(0)"  maschine kl
ein     $  20000"                        <mo>
11450 printspc(5)"(typ a, 1 ware p
ro monat)                                <ac>
11460 print                              <hk>
11470 print" "ma$(1)"  maschine mi
ttel    $  60000"                        <do>
11480 printspc(5)"(typ b, 2 waren
pro monat)"                              <ho>
11490 print                              <lf>
11500 print" "ma$(2)"  maschine gr
oss     $ 100000"                        <fk>
11510 printspc(5)"(typ c, 3 waren
pro monat)"                              <kh>
11520 print                              <pd>
11530 print"  "ab$"  arbeiter
     $   5000"                           <ik>
11540 printspc(5)"(preis fuer eins
chulung)"                                <jd>
11550 print:print:printspc(9)"kapi
tal: $"gd(0)+gw(0)                       <po>
11560 sysu9,2,20,7,22,0:printrv$"e
nde"                                     <om>
11570 a=-1:sysu6                         <pn>
11580 sysu7,2,20,7,22,12100              <pf>
11590 fori=0to4                          <ed>
11600 sysu7,2,5+i*3,3,6+i*3,11620        <ha>
11610 goto11630                          <bk>
11620 a=i:i=4                            <gc>
11630 next                               <lj>
11640 ifa=-1then11570                    <je>
11650 sysu5,0,1                          <ai>
11660 restore15830+a*10                  <fe>
11670 sysu9,12,5,28,21,2:scnclr          <ph>
11680 reada$:x=15:gosub2430              <me>
11690 sysu9,14,9,26,11,0                 <km>
11700 ifa=4thenprintrv$"einstellun
g":elseprintrv$"  einkauf  "             <hj>
11710 sysu9,14,13,26,15,0                <ce>
11720 ifa=4thenprintrv$"entlassung
 ":elseprintrv$"  verkauf  "             <nk>
11730 sysu9,14,17,26,19,0:printrv$
"  zurueck  "                            <lm>
11740 sysu6                              <kd>
11750 sysu7,14,9,26,11,11790             <gi>
11760 sysu7,14,13,26,15,11940            <bi>
11770 sysu7,14,17,26,19,12060            <id>
11780 goto11740                          <lp>
11790 ifa=0andcp(0)=1then11740           <no>
11800 ifa=4andab(0)>59then11740          <gd>
11810 ifa>0anda<4andma>59then11740       <fp>
11820 sysu9,14,9,26,11,1                 <co>
11830 ifa=4thenprint"einstellung":
elseprint"  einkauf  "                   <mp>
11840 ifa=0theny=150000:cp(0)=1          <mc>
11850 ifa=1theny=20000:ma(0,0)=ma(
0,0)+1:ma=ma+1                           <fc>
11860 ifa=2theny=50000:ma(0,1)=ma(
0,1)+1:ma=ma+1                           <gn>
11870 ifa=3theny=100000:ma(0,2)=ma
(0,2)+1:ma=ma+1                          <lm>
11880 ifa=4theny=5000:ab(0)=ab(0)+
1                                        <pc>
11890 gw(0)=gw(0)-y                      <dh>
11900 sysu4,30                           <km>
11910 sysu5,1,0                          <ge>
11920 char,20,21,str$(gd(0)+gw(0))
+".      "                               <bl>
11930 goto11570                          <ep>
11940 ifa=0andcp(0)=0then11740           <mn>
11950 ifa=4andab(0)=0then11740           <jf>
11960 ifa>0anda<4thenifma(0,a-1)=0
then11740                                <ik>
11970 sysu9,14,13,26,15,1                <dm>
11980 ifa=4thenprint"entlassung ":
elseprint"  verkauf  "                   <ig>
11990 ifa=0theny=75000:cp(0)=0           <lf>
12000 ifa=1theny=10000:ma(0,0)=ma(
0,0)-1                                   <ld>
12010 ifa=2theny=25000:ma(0,1)=ma(
0,1)-1                                   <fc>
12020 ifa=3theny=50000:ma(0,2)=ma(
0,2)-1                                   <dh>
12030 ifa=4theny=-2*lh(0):ab(0)=ab
(0)-1                                    <oo>
```

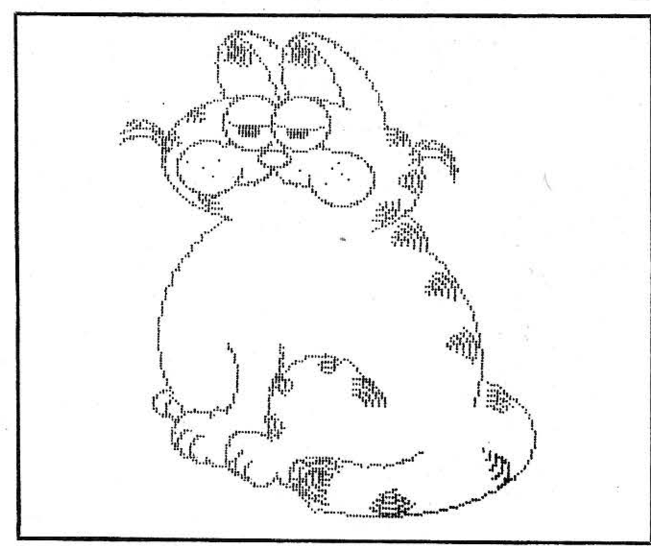
```
12040 gw(0)=gw(0)+y                          <cm>
12050 goto11900                              <hd>
12060 sysu9,14,17,26,19,1:print"
zurueck  "                                   <li>
12070 sysu4,40                               <ef>
12080 sysu5,1,0                              <kd>
12090 goto11570                              <hd>
12100 sysu9,2,20,7,22,1:print"ende
"                                            <ph>
12110 sysu4,40                               <fd>
12120 :                                      <ej>
12130 rem  dauerzahlungen                    <oi>
12140 :                                      <hb>
12150 sysu9,0,0,39,24,2:scnclr               <ao>
12160 a$=" d a u e r z a h l u n g
 e n ":x=38:gosub2430                        <lg>
12170 print:print                            <dn>
12180 restore15880                           <jo>
12190 x(0)=we(0):x(1)=fo(0):x(2)=v
g(0)*500:x(3)=lh(0)                          <bb>
12200 fori=0to3                              <gd>
12210 sysu9,6,5+i*4,8,7+i*4,0:prin
trv$"-"                                      <gj>
12220 sysu9,31,5+i*4,33,7+i*4,0:pr
intrv$"+"                                    <il>
12230 reada$                                 <go>
12240 sysu9,10,5+i*4,29,7+i*4,0:pr
intusingrv$+a$+": $######";x(i)              <ef>
12250 next                                   <je>
12260 sysu9,17,21,22,23,0:printrv$
"ende"                                       <og>
12270 sysu6:i=0                              <ko>
12280 sysu7,17,21,22,23,12430                <fl>
12290 sysu7,6,5+i*4,8,7+i*4,12330            <od>
12300 sysu7,31,5+i*4,33,7+i*4,1238
0                                            <ph>
12310 i=i+1:ifi<=azthen12290                 <ka>
12320 goto12270                              <db>
12330 x(i)=x(i)-500-(i=3)*450                <gp>
12340 ifx(i)<0thenx(i)=0                     <ae>
12350 char,23,6+i*4,""                       <la>
12360 printusingrv$+"######";x(i)            <nj>
12370 goto12270                              <nn>
12380 x(i)=x(i)+500+(i=3)*450                <ee>
12390 ifx(i)>500000thenx(i)=500000           <ik>
12400 char,23,6+i*4,""                       <om>
12410 printusingrv$+"######";x(i)            <ak>
12420 goto12270                              <fi>
12430 sysu9,17,21,22,23,1:print"en
de"                                          <ek>
12440 we(0)=x(0):fo(0)=x(1):vg(0)=
x(2)/500:lh(0)=x(3)                          <hl>
12450 fori=0to2                              <ci>
12460 gw(0)=gw(0)-x(i)                       <oa>
12470 next                                   <ep>
12480 b=0                                    <cb>
12490 fori=0toaz                             <km>
12500 ifsk(i)thenb=b+ab(i)                   <aj>
12510 fw(i)=fw(i)+z(i):iffw(i)>60t
henfw(i)=60:ig(i)=ig(i)*.97                  <lj>
12520 next                                   <lc>
12530 fw%=fw%-int(sqr(gw/1000))              <bl>
12540 sysu2                                  <lg>
12550 fori=1toaz                             <be>
12560 e=ma(i,0)+ma(i,1)*2+ma(i,2)*
3                                            <bk>
12570 ifsk(i)=0then12670                     <gb>
12580 d=e*(fp-rp-pk(i)/(cp(i)+1))+
(sk(i)+500)*ab(i)                            <al>
12590 ifd<100then12670                       <ie>
12600 x=gd(i)+gw(i)-sc(i)-dg                 <eg>
12610 ifx>dande/dp>1.1then12670              <ng>
12620 ifkp<am*500andb>=azthen12670           <nj>
12630 y=e1(i)/2+sgn(x)*3+3*e/dp-sq
r(d/e/50)+dl/lh(i)*5-cp(i)*3+b/am*
5                                            <ho>
12640 ifti/da<1theny=y+ti/da*3               <kn>
12650 ify>9ory>(1600-lh(i))/30then
12670                                        <lc>
12660 lh(i)=lh(i)+50                         <pe>
12670 ifsk(i)thenwe(i)=0:goto12770           <ed>
12680 x=gw/(az+1)*.8                         <nl>
12690 ifx<1000thenx=x+500                    <on>
12700 x=2500/fp*x                            <mn>
12710 iffp>3500orfw<fw%-2thenx=x*.
7                                            <ll>
12720 x=x*e/dp:iffw-fw%>5thenx=x+1
00                                           <mn>
12730 we(i)=we(i)+(x-we(i))*.3               <im>
12740 iffw-fw%>5thenwe(i)=we(i)*1
1+50                                         <hc>
12750 ifwe(i)>x*1.5thenwe(i)=we(i)
*.8                                          <af>
12760 iffw(i)>e*3orfw(i)>50thenwe(
i)=we(i)*1.2+1000                            <cm>
12770 we(i)=int(we(i)):gw(i)=gw(i)
-we(i)                                       <bj>
12780 ifpk(i)<300+20*e2(i)thenfo(i
)=0:goto12820                                <ho>
12790 x=int((10-e2(i))*200000/(pk(
i)-200))                                     <na>
12800 ifti/da>.9-e2(i)/50thenfo(i)
=0:goto12820                                 <ph>
12810 fo(i)=x                                <mg>
12820 ifrnd(0)*x*(pk(i)-200)<50000
0orpk<300orrnd(1)<.8then12840                <ii>
12830 pk(i)=pk(i)-50                         <do>
12840 ify(i)>0andfw(i)>vg(i)andfp-
y(i)>1000thenvg(i)=vg(i)+1                   <io>
12850 iffw(i)<vg(i)or(y(i)>0andy(i
)+500<fp)thenvg(i)=vg(i)-1:ifvg(i)
<0thenvg(i)=0                                <ea>
12860 gw(i)=gw(i)-vg(i)*500                  <jp>
12870 ifcp(i)then12890                       <jj>
12880 ifgd(i)+gw(i)>125000-e3(i)an
dti/da<.8thencp(i)=1:gw(i)=gw(i)-1
```

```
50000                                   <ap>
12890 ifma(i,0)+ma(i,1)+ma(i,2)>19
then12980                               <ch>
12900 x=int(dp/db-e2(i)/7):x(0)=20
000:x(1)=50000:x(2)=100000:ifx>2th
enx=2                                   <an>
12910 ifgd(i)+gw(i)<(10*x(x))/(10+
e3)+lh(i)*(e4(i)/3+1)then12980          <pd>
12920 ma(i,x)=ma(i,x)+1:ab(i)=ab(i
)+1:gw(i)=gw(i)-x(x)-5000               <ao>
12930 x=gw(i):ifx<5000then12980         <mh>
12940 gw(i)=gw(i)-(x-5000)*.2:ifx<
10000then12980                          <jp>
12950 gw(i)=gw(i)-(x-10000)*.1:ifx
<20000then12980                         <fg>
12960 gw(i)=gw(i)-(x-20000)*.1:ifx
<50000then12980                         <kg>
12970 gw(i)=gw(i)-(x-50000)*.2          <db>
12980 gd(i)=gd(i)+gw(i)                 <jl>
12990 if(gd(i)>999andsc(i)>0)orgd(
i)<0thengd(i)=gd(i)-1000*int(gd(i)
/1000)                                  <dn>
13000 if(gd(i)>999andsc(i)>0)orgd(
i)<0thensc(i)=sc(i)-1000*int(sc(i)
/1000)                                  <ne>
13010 x=ma(i,0)*10000+ma(i,1)*2500
0+ma(i,2)*50000+cp(0)*75000+10000       <fb>
13020 ifgw(i)<1000thenig(i)=ig(i)*
.99:elseig(i)=ig(i)+(1-ig(i))*.01       <po>
13030 ifsc(i)>xthensc(i)=x              <pi>
13040 ifkp<0andlh(i)>1000thenlh(i)
=lh(i)-50                               <mf>
13050 next                              <nk>
13060 :                                 <ki>
13070 rem  monatsbilanz                 <ic>
13080 :                                 <na>
13090 sysu9,0,0,39,24,2:scnclr          <mi>
13100 a$=" b i l a n z ":x=38:gosu
b2430                                   <kp>
13110 print:print                       <lj>
13120 c=ma(0,0)*10000+ma(0,1)*2500
0+ma(0,2)*50000+cp(0)*75000+10000       <hd>
13130 a=gw(0):ifa<5000then13180         <hj>
13140 gw(0)=gw(0)-(a-5000)*.2:ifa<
10000then13180                          <mb>
13150 gw(0)=gw(0)-(a-10000)*.1:ifa
<20000then13180                         <hp>
13160 gw(0)=gw(0)-(a-20000)*.1:ifa
<50000then13180                         <ji>
13170 gw(0)=gw(0)-(a-50000)*.2          <mm>
13180 gw(0)=int(gw(0)):d=gd(0)+gw(
0):e=sc(0)                              <be>
13190 sysu3                             <cb>
13200 if(d>999ande>999)ord<0thend=
d-1000*int(d/1000):e=e-(gd(0)+gw(0
)-d)                                    <dg>
13210 ife<dthend=d-e:e=0                <eh>
13220 ife>cthen15180                    <pj>
13230 print" vor einem monat:"          <ca>
13240 print                             <ha>
13250 printusing"   kapital  : $###
####";gd(0)                             <oa>
13260 printusing"   schulden: $###
####";sc(0)                             <na>
13270 print                             <kl>
13280 print" in diesem monat:"          <ja>
13290 print                             <nd>
13300 printusing"   kapital  : $###
####";d                                 <fk>
13310 printusing"   schulden: $###
####";e                                 <do>
13320 print                             <bb>
13330 print" sie sind kreditwuerdi
g bis $"c;le$"."                        <ej>
13340 print                             <dj>
13350 print" sie zahlen $"a-gw(0)"
steuern."                               <ne>
13360 gd(0)=d:sc(0)=e                   <op>
13370 ifti+1800<dathen13610             <do>
13380 gosub3270                         <na>
13390 y=d-e+(c-10000)                   <mo>
13400 printup$up$up$ec$"t"cl$;          <pf>
13410 print" ihre spielzeit ist nu
n abgelaufen."                          <fn>
13420 print" nach verkauf aller sa
chwerte bleiben"                        <di>
13430 print" ihnen nach"mo"monaten
 $"y;le$", die"                         <kb>
13440 print" sie aus $ 25000 erwir
tschaftet haben."                       <oe>
13450 print                             <bg>
13460 y=(y-25000)*5000/da*80/(mo+1
0)                                      <dk>
13470 ify<0thena$="miserabler":got
o13540                                  <go>
13480 ify<500thena$="schlechter":g
oto13540                                <fp>
13490 ify<1000thena$="maessiger":g
oto13540                                <dl>
13500 ify<2000thena$="passabler":g
oto13540                                <gk>
                                        <mi>
13510 ify<4000thena$="guter":goto1
3540                                    <mi>
13520 ify<10000thena$="sehr guter"
:goto13540                              <nd>
13530 a$="exzellenter"                  <kg>
13540 print" sie waren ein "a$" ma
nager."                                 <cp>
13550 sysu4,100                         <dj>
13560 gosub3270                         <fl>
13570 restore15470:gosub2600            <nb>
13580 ifx=1thenrun                      <fa>
13590 printec$"]"ec$"n"                 <do>
13600 end                               <ck>
13610 gosub3270                         <lf>
```

```
13620 sysu5,0,2                                  <kc>
13630 sysu9,14,6,25,20,2:scnclr                  <nm>
13640 sysu9,15,7,19,10,0:printrv$g
1$                                               <en>
13650 sysu9,20,7,24,10,0:printrv$g
2$                                               <cc>
13660 sysu9,15,11,24,13,0:printrv$
" weiter "                                       <da>
13670 sysu9,15,14,24,16,0:printrv$
"  ende   "                                      <io>
13680 sysu9,15,17,24,19,0:printrv$
"  save   "                                      <ad>
13690 sysu6                                      <ng>
13700 sysu7,15,7,19,10,13760                     <ce>
13710 sysu7,20,7,24,10,14060                     <ai>
13720 sysu7,15,11,24,13,14400                    <pj>
13730 sysu7,15,14,24,16,14430                    <oc>
13740 sysu7,15,17,24,19,14490                    <cp>
13750 goto13690                                  <cl>
13760 sysu5,0,1                                  <pl>
13770 sysu9,15,7,19,10,1:printg1$                <ko>
13780 sysu4,40                                   <fe>
13790 sysu9,2,3,37,17,2:scnclr                   <ea>
13800 a$="einnahmen - ausgaben (in
 $)":x=34:gosub2430                              <il>
13810 a$=" "+left$(li$,32)                       <gg>
13820 a=-999999:b=999999                         <df>
13830 fori=0toaz                                 <jd>
13840 ifgw(i)>athena=gw(i)                       <ka>
13850 ifgw(i)<bthenb=gw(i)                       <na>
13860 next                                       <dg>
13870 a=1000*int(a/1000+1)                       <ig>
13880 c=1000*int((a-b)/4000+1)                   <pc>
13890 b=a-c*4                                    <me>
13900 fori=0to4                                  <co>
13910 printa$:printusing" "+rv$+"+
######"+ro$+chr$(186);a-c*i                      <oe>
13920 next                                       <kp>
13930 printspc(8);                               <eh>
13940 fori=0toaz                                 <og>
13950 print" "fb$(i);                            <ag>
13960 next                                       <pp>
13970 y=14                                       <le>
13980 forc=0toaz                                 <aj>
13990 x=12+c*4                                   <ga>
14000 z=int((gw(c)-b)/(a-b)*64+.5)               <pf>
14010 gosub2840                                  <oj>
14020 next                                       <hh>
14030 gosub3270                                  <fi>
14040 sysu5,1,0                                  <fl>
14050 goto13690                                  <fl>
14060 sysu5,0,1                                  <do>
14070 sysu9,20,7,24,10,1:printg2$                <ib>
14080 sysu4,40                                   <lj>
14090 sysu9,3,4,37,22,2:scnclr                   <jd>
14100 a$="inventur":x=33:gosub2430               <ia>
14110 print                                      <ed>
14120 c=0:ifcp(0)thenprint:print"
";cp$:c=1                                        <ib>
14130 ifma=0then14220                            <bk>
14140 fori=2to0step-1                            <gd>
14150 ifma(0,i)=0then14210                       <mi>
14160 forz=1toma(0,i)                            <ma>
14170 y=int(c/7):x=c-y*7                         <an>
14180 char,x*5+6,y*3+9,ma$(i)                    <ha>
14190 c=c+1                                      <ap>
14200 next                                       <ob>
14210 next                                       <pf>
14220 ifab(0)=0then14270                         <ni>
14230 fori=0+cp(0)toab(0)-1+cp(0)                <gn>
14240 y=int(i/7):x=i-y*7                         <gb>
14250 char,x*5+4,y*3+9,ab$                       <kf>
14260 next                                       <fj>
14270 ifrw(0)<15theni=21:goto14310               <gi>
14280 fori=21to22-int(rw(0)/15)ste
p-1                                              <jf>
14290 char,4,i,rp$                               <bg>
14300 next                                       <ki>
14310 char,4,i,left$(rp$,rw(0)-(15
*(21-i)))                                        <ph>
14320 iffw(0)<15theni=21:goto14360               <kc>
14330 fori=21to22-int(fw(0)/15)ste
p-1                                              <co>
14340 char,22,i,fp$                              <jl>
14350 next                                       <ao>
14360 char,22,i,left$(fp$,fw(0)-(1
5*(21-i)))                                       <ln>
14370 gosub3270                                  <be>
14380 sysu5,2,0                                  <cb>
14390 goto13690                                  <mm>
14400 sysu9,15,11,24,13,1:print" w
eiter "                                          <kh>
14410 sysu4,40                                   <la>
14420 goto3320                                   <ca>
14430 sysu9,15,14,24,16,1:print"
ende   "                                         <jo>
14440 sysu4,40                                   <cl>
14450 restore15490:gosub2600                     <kd>
14460 ifx=0thensysu9,15,14,24,16,0
:printrv$"  ende   ":goto13690                   <mm>
14470 printec$"n"ec$"l"                          <gp>
14480 end                                        <bb>
14490 sysu5,0,1                                  <hc>
14500 sysu9,15,17,24,19,1:print"
save   "                                         <ol>
14510 sysu4,40                                   <jn>
14520 sysu9,9,8,30,16,2:scnclr                   <aa>
14530 a$="speichermedium ?":x=20:g
osub2430                                         <ee>
14540 sysu9,12,12,17,14,0:printrv$
"tape"                                           <hk>
14550 sysu9,22,12,27,14,0:printrv$
"disk"                                           <dl>
14560 sysu6                                      <bl>
14570 sysu7,12,12,17,14,14600                    <je>
14580 sysu7,22,12,27,14,14630                    <ff>
```

```
14590 goto14560                         <na>
14600 sysu9,12,12,17,14,1:print"ta
pe"                                     <im>
14610 sysu4,40                          <mg>
14620 c=1:goto14660                     <op>
14630 sysu9,22,12,27,14,1:print"di
sk"                                     <cm>
14640 sysu4,40                          <dm>
14650 c=8                               <ea>
14660 sysu9,9,8,30,16,2:scnclr          <al>
14670 a$="dateiname ?":x=20:gosub2
430                                     <eb>
14680 print:print                       <aa>
14690 print" >";:x=16:gosub2270:c$
=a$                                     <hb>
14700 sysu4,20                          <gj>
14710 sysu9,9,8,30,16,2:scnclr          <ho>
14720 a$="  legen sie die    ":x=20
:gosub2430                              <of>
14730 print" "rv$"  ";                  <hc>
14740 ifc=1thenprint"kassette";:el
seprint"diskette";                      <pn>
14750 print" ein     "                  <dj>
14760 sysu9,17,13,22,15,0:printrv$
" ok "                                  <om>
14770 sysu6                             <le>
14780 sysu7,17,13,22,15,14800           <kp>
14790 goto14770                         <eb>
14800 sysu9,17,13,22,15,1:print" o
k "                                     <ah>
14810 b$=ti$                            <oe>
14820 ifc=8thenc$=c$+",s,w"             <de>
14830 printbk$;:open1,c,c,c$:print
wt$;                                    <pn>
14840 gosub3200:ifxthen15090            <on>
14850 sysu9,9,8,30,16,2:scnclr          <je>
14860 a$=" saving... ":x=20:gosub2
430                                     <fd>
14870 print:print:x=0                   <op>
14880 fori=0to5                         <af>
14890 print#1,na$(i)","fb$(i)","rw
(i)","fw(i)","ma(i,0)","ma(i,1)","
ma(i,2)                                 <oe>
14900 gosub3200:ifxthen14950            <jb>
14910 print#1,cp(i)","ab(i)","gd(i
)","sc(i)","lh(i)","pk(i)","ia(i)"
,"ik(i)","ig(i)                         <pa>
14920 gosub3200:ifxthen14950            <jc>
14930 print#1,vg(i)","sk(i)","we(i
)","fo","e1(i)","e2(i)","e3(i)","e
4(i)                                    <ge>
14940 gosub3200                         <jo>
14950 ifxtheni=5                        <bb>
14960 next                              <nh>
14970 ifxthen15090                      <ef>
14980 print#1,rp","fp","kp","rw","
fw","rw%","fw%","na$","da","da$         <kk>
14990 gosub3200:ifxthen15090            <cd>
15000 print#1,gr","gf","f1","f2","
f3","mo","ti$                           <me>
15010 gosub3200:ifxthen15090            <bi>
15020 close1                            <al>
15030 gosub3200:ifxthen15090            <bn>
15040 scnclr:a$=" fertig ":x=20:go
sub2430                                 <no>
15050 ti$=b$                            <kc>
15060 sysu4,100                         <fd>
15070 sysu5,1,0                         <eo>
15080 goto13690                         <ob>
15090 close1                            <cc>
15100 sysu9,9,8,30,16,2:scnclr          <ni>
15110 print:print:print:print:prin
tspc(3)"schreibfehler"                  <ig>
15120 a$="schreibfehler":x=20:gosu
b2430                                   <ol>
15130 close1                            <mf>
15140 ti$=b$                            <gh>
15150 sysu4,300                         <ke>
15160 sysu5,1,0                         <mf>
15170 goto13690                         <jg>
15180 print:print                       <la>
15190 print" an"                        <jo>
15200 print                             <na>
15210 print" herrn/frau/fraeulein
"na$                                    <bc>
15220 print:print                       <op>
15230 print" ich bedauere ihnen mi
tteilen zu"                             <kf>
15240 print" mussen, dass ihre fir
ma "na$(0)                              <jd>
15250 print" in konkurs gegangen i
st."                                    <ob>
15260 print" die bank verlangt die
 sofortige"                             <hk>
15270 print" zurueckzahlung des kr
edites in der"                          <ln>
15280 print" hoehe von $"e;le$"."       <hn>
15290 print" ihr besitz wird verst
eigert."                                <hp>
15300 gosub3270                         <mj>
15310 restore15470:gosub2600            <pe>
15320 ifx=0thenprintec$"n"ec$"l":e
nd                                      <fk>
15330 run                               <me>
15340 :                                 <ii>
15350 rem  entscheidungsfragen          <fh>
15360 :                                 <lc>
15370 data3                             <fb>
15380 data" wollen sie die polizei
"                                       <mh>
15390 data"   auffordern haerter
"                                       <fn>
15400 data"    durchzugreifen ?
"                                       <cp>
15410 data2                             <pa>
15420 data" wollen sie den auftrag
```

```
"                                   <gc>
15430 data"        annehmen ?
"                                   <oh>
15440 data2                         <gl>
15450 data" wollen sie eine studie
"                                   <ed>
15460 data"   in auftrag geben ?
"                                   <pi>
15470 data1                         <oc>
15480 data"    noch ein spiel ?
"                                   <cp>
15490 data1                         <db>
15500 data"   sind sie sicher ?
"                                   <bk>
15510 data2                         <ic>
15520 data"      wollen sie die
"                                   <hd>
15530 data"   maschine kaufen ?
"                                   <ph>
15540 data2                         <pk>
15550 data"  wollen sie den mann
"                                   <on>
15560 data"       einstellen ?
"                                   <fo>
15570 data2                         <hb>
15580 data" wollen sie das angebot
"                                   <lb>
15590 data"        annehmen ?
"                                   <kn>
15600 data1                         <oi>
15610 data"  wollen sie zahlen ?
"                                   <ll>
15620 :                             <lm>
15630 rem  zahleneingabe            <ji>
15640 :                             <oe>
15650 data2                         <le>
15660 data" wieviele waren wollen
"                                   <bl>
15670 data"   sie produzieren ?
"                                   <mk>
15680 data2                         <cm>
15690 data" wieviele waren wollen
"                                   <ff>
15700 data"    sie ersteigern ?
"                                   <mn>
15710 data12                        <hp>
15720 data"   wieviel wollen sie
"                                   <fh>
15730 data"     maximal bieten ?
"                                   <ml>
15740 data2                         <bp>
15750 data" wieviele waren wollen
"                                   <mj>
15760 data"      sie anbieten ?
"                                   <dn>
15770 data2                         <jg>
15780 data"   wieviel wollen sie
"                                   <ol>
15790 data" mindestens verlangen ?
"                                   <bb>
15800 :                             <cf>
15810 rem  sachwerte und personal   <no>
15820 :                             <em>
15830 data"computer"                <pn>
15840 data"maschine a"              <fe>
15850 data"maschine b"              <li>
15860 data"maschine c"              <jf>
15870 data"arbeiter"                <eo>
15880 data"werbung  "               <bp>
15890 data"forschung"               <fa>
15900 data"vertrieb "               <nl>
15910 data"loehne   "               <nb>
15920 :                             <bg>
15930 rem  datas                    <fd>
15940 :                             <dp>
15950 data,,,,,,,,,25000,,1000,100
0,.5,.5,.5,,,,,                     <fm>
15960 datacommodore,com,43,19,1,2,
2,1,5,132400,,1200,800,.4,.7,.8,2,
6,3,6,4                             <ko>
15970 datasonny,son,6,12,,2,,,2,13
800,,1000,900,.7,.2,.8,,3,2,8,8     <om>
15980 datamazda,maz,3,,,1,,,1,2700
,20000,1000,1000,.5,.9,.2,,9,,6,2   <he>
15990 datamiele,mil,28,12,4,1,2,1,
7,3400,50000,1100,900,.2,.7,.4,2,8
,0,9,5                              <ap>
16000 dataphillips,pip,12,11,,2,,,
2,125900,,1300,1000,.9,.5,,1,1,9,1
,7                                  <ko>
16010 rem ======================    <eh>
16020 rem 60671 bytes memory        <nm>
16030 rem 06768 bytes m-code        <nn>
16040 rem 38517 bytes program       <nb>
16050 rem 00518 bytes variables     <ph>
16060 rem 01072 bytes arrays        <lp>
16070 rem 00744 bytes strings       <lk>
16080 rem 13052 bytes free (0)      <dp>
16090 rem ======================    <kf>
```



MANAGER-STARTPROGRAMM============p4
(p) COMMODORE WELT TEAM
======================================
(c) by Bernhard Oemer

Plus4 (C16/116 + 64 KB)
======================================
Das Programm ist mit dem Maschinensprachmonitor TEDMON unter Zuhilfenahme des Pruefsummenprogrammes CHECKMON einzugeben und auf Diskette mit s"manager",8,1001,1750 abzuspeichern. Auf Kassette bitte mit s"manager",1,1001,1750 abspeichern. Geladen und gestartet wird wie bei einem normalen Basic-Progamm.
======================================

```
>1000 00 14 10 0a 00 e7 30 2c :<c7>
>1008 31 3a e7 34 2c 31 3a e7 :<ed>
>1010 31 2c 31 00 2a 10 14 00 :<27>
>1018 58 24 b2 c7 28 31 33 29 :<eb>
>1020 3a 52 24 b2 c7 28 33 34 :<ad>
>1028 29 00 6d 10 1e 00 41 24 :<c3>
>1030 b2 22 53 d9 34 32 35 36 :<da>
>1038 22 aa 58 24 aa 22 4c cf :<4e>
>1040 22 aa 52 24 aa 22 4d 41 :<02>
>1048 4e 41 47 45 52 2e 32 22 :<1c>
>1050 aa 52 24 aa 22 2c 22 aa :<e0>
>1058 c4 28 c2 28 31 37 34 29 :<b8>
>1060 29 aa 58 24 aa 22 52 d5 :<79>
>1068 22 aa 58 24 00 88 10 28 :<df>
>1070 00 f9 31 2c 41 24 3a 97 :<2b>
>1078 32 30 33 35 2c 30 3a 9e :<87>
>1080 35 36 33 36 34 3a a2 00 :<44>
>1088 00 00 97 31 33 32 31 2c :<a0>
>1090 31 33 00 99 10 50 00 a2 :<67>
>1098 00 00 00 00 00 00 00 00 :<41>
>10a0 a2 00 bd 00 d0 9d 00 f8 :<82>
>10a8 bd 00 d1 9d 00 f9 bd 70 :<50>
>10b0 15 9d 00 fa bd 70 16 9d :<0e>
>10b8 00 fb ca d0 e5 78 a9 18 :<d0>
>10c0 8d 14 03 a9 12 8d 15 03 :<11>
>10c8 58 a9 00 85 d5 a9 f8 8d :<be>
>10d0 13 ff a9 c0 8d 12 ff a9 :<8a>
>10d8 e8 85 38 a9 15 85 2c a9 :<3a>
>10e0 71 85 2b a9 00 8d 70 15 :<7c>
>10e8 20 7b 8a 4c 0a 80 20 e1 :<8c>
>10f0 11 20 91 94 20 d2 9d a5 :<1c>
>10f8 14 85 d8 a5 15 85 d9 86 :<3e>
>1100 d6 a9 00 85 d7 a5 d6 4a :<09>
>1108 4a 4a aa 20 20 12 a5 d9 :<c5>
>1110 4a a5 d8 6a 4a 4a a8 b1 :<d9>
>1118 d0 8d 5e 06 a9 7c 91 d0 :<f6>
>1120 c8 b1 d0 8d 60 06 a9 7e :<06>
>1128 91 d0 98 18 69 27 a8 b1 :<70>
>1130 d0 8d 5f 06 a9 7d 91 d0 :<66>
>1138 c8 b1 d0 8d 61 06 a9 7f :<49>
>1140 91 d0 78 8d 3f ff ad 5e :<78>
>1148 06 a2 7c 20 2b 12 ad 5f :<db>
>1150 06 a2 7d 20 2b 12 ad 60 :<2e>
>1158 06 a2 7e 20 2b 12 ad 61 :<c2>
>1160 06 a2 7f 20 2b 12 a2 3f :<13>
>1168 bd 65 12 9d 62 06 ca 10 :<e8>
>1170 f7 a5 d8 29 07 f0 18 85 :<91>
>1178 d4 a2 08 a4 d4 5e 62 06 :<da>
>1180 7e 72 06 5e 82 06 7e 92 :<41>
>1188 06 88 10 f1 ca 10 ec a5 :<7d>
>1190 d6 29 07 f0 30 18 69 08 :<18>
>1198 aa a0 08 b9 62 06 9d 62 :<48>
>11a0 06 b9 72 06 9d 72 06 b9 :<bb>
>11a8 82 06 9d 82 06 b9 92 06 :<59>
>11b0 9d 92 06 a9 00 99 62 06 :<08>
>11b8 99 72 06 99 82 06 99 92 :<7b>
>11c0 06 ca 88 10 d6 a2 1f bd :<02>
>11c8 62 06 49 ff 3d e0 fb 1d :<46>
>11d0 82 06 9d e0 fb ca 10 ef :<16>
>11d8 a9 ff 85 d5 8d 3e ff 58 :<af>
>11e0 60 24 d5 30 01 60 a9 00 :<b6>
>11e8 85 d5 a5 d6 4a 4a 4a aa :<67>
>11f0 20 20 12 a5 d9 4a a5 d8 :<bc>
>11f8 6a 4a 4a a8 ad 5e 06 91 :<47>
>1200 d0 c8 ad 60 06 91 d0 18 :<9b>
>1208 98 69 27 a8 ad 5f 06 91 :<67>
>1210 d0 c8 ad 61 06 91 d0 60 :<e6>
>1218 a9 c0 8d 12 ff 4c 0e ce :<11>
>1220 bd 02 d8 85 d0 bd 1b d8 :<c6>
>1228 85 d1 60 48 20 4d 12 a5 :<ab>
>1230 d0 85 d2 a5 d1 85 d3 8a :<f9>
>1238 20 4d 12 a0 07 68 aa b1 :<43>
>1240 d2 e0 00 10 02 49 ff 91 :<56>
>1248 d0 88 10 f3 60 29 7f a0 :<74>
>1250 00 84 d1 0a 26 d1 0a 26 :<61>
>1258 d1 0a 26 d1 85 d0 a5 d1 :<9a>
>1260 69 f8 85 d1 60 fc fc f8 :<38>
>1268 fc fe df 0f 07 02 00 00 :<83>
>1270 00 00 00 00 00 00 00 00 :<f2>
>1278 00 00 00 80 00 00 00 00 :<06>
>1280 00 00 00 00 00 00 78 70 :<69>
>1288 78 5c 0e 07 02 00 00 00 :<e2>
>1290 00 00 00 00 00 00 00 00 :<33>
>1298 00 00 00 00 00 00 00 00 :<43>
>12a0 00 00 00 00 00 ad 06 ff :<e9>
>12a8 29 ef 8d 06 ff 60 ad 06 :<fa>
>12b0 ff 09 10 8d 06 ff 60 20 :<15>
>12b8 de 9d a5 14 d0 05 a5 15 :<61>
>12c0 d0 01 60 a5 15 30 35 a9 :<85>
>12c8 00 85 d4 78 a9 ed 8d 14 :<17>
>12d0 03 a9 12 8d 15 03 58 a5 :<d0>
>12d8 d4 d0 05 20 e1 ff d0 f7 :<ab>
>12e0 78 a9 18 8d 14 03 a9 12 :<3e>
>12e8 8d 15 03 58 60 c6 14 d0 :<0d>
>12f0 08 c6 15 10 04 a9 ff 85 :<b1>
>12f8 d4 4c 0e ce 4c 1c 99 20 :<c4>
>1300 91 94 20 84 9d e0 03 b0 :<50>
>1308 f3 86 d4 20 d8 9d 8a c9 :<ad>
>1310 03 b0 e9 c5 d4 f0 e5 85 :<98>
```

```
>1318 db c9 00 d0 03 20 72 14 :<09>
>1320 a2 0b b5 c2 9d e8 0f ca :<5f>
>1328 10 f8 a2 0c bd e5 07 9d :<d6>
>1330 f4 0f ca 10 f7 a9 00 85 :<f3>
>1338 d0 85 d2 a6 d4 bd 7b 13 :<16>
>1340 85 d1 a6 db bd 7b 13 85 :<ce>
>1348 d3 78 8d 3f ff a2 07 a0 :<c9>
>1350 00 b1 d0 91 d2 c8 d0 f9 :<89>
>1358 e6 d1 e6 d3 ca 10 f0 8d :<5a>
>1360 3e ff 58 a2 0c bd f4 0f :<53>
>1368 9d e5 07 ca 10 f7 a2 0b :<84>
>1370 bd e8 0f 95 c2 ca 10 f8 :<bf>
>1378 4c a8 d8 08 e8 f0 00 04 :<bc>
>1380 08 00 01 05 09 00 02 06 :<5c>
>1388 0a 01 01 00 01 00 00 00 :<44>
>1390 01 00 00 20 e1 11 20 05 :<e0>
>1398 11 a2 fd 78 8e 08 ff ad :<40>
>13a0 08 ff 8e 08 ff cd 08 ff :<db>
>13a8 d0 f2 58 49 ff a8 29 0f :<7a>
>13b0 aa bd 7e 13 c0 0f 90 05 :<6e>
>13b8 a9 00 85 ef 60 c9 00 f0 :<4e>
>13c0 d8 85 db 20 72 14 20 e1 :<f4>
>13c8 11 20 e2 13 20 e2 13 a6 :<6c>
>13d0 db bd 88 13 f0 03 20 e2 :<1c>
>13d8 13 ea ea ea 20 05 11 4c :<9f>
>13e0 99 13 a5 db 29 01 f0 10 :<b4>
>13e8 a5 d8 05 d9 f0 0a c6 d8 :<b2>
>13f0 a5 d8 c9 ff d0 02 c6 d9 :<83>
>13f8 a5 db 29 02 f0 10 a5 d9 :<2a>
>1400 f0 06 a5 d8 c9 37 b0 06 :<c7>
>1408 e6 d8 d0 02 e6 d9 a5 db :<5d>
>1410 29 04 f0 06 a5 d6 f0 02 :<57>
>1418 c6 d6 a5 db 29 08 f0 08 :<b2>
>1420 a5 d6 c9 bf b0 02 e6 d6 :<5d>
>1428 60 20 d8 9d 86 d0 20 d8 :<a7>
>1430 9d 86 d1 20 d8 9d 86 d2 :<fe>
>1438 20 d8 9d 86 d3 20 91 94 :<d6>
>1440 a5 d9 4a a5 d8 6a 4a 4a :<77>
>1448 c5 d0 90 20 aa ca e4 d2 :<1e>
>1450 b0 1a a5 d6 4a 4a 4a c5 :<bd>
>1458 d1 90 11 aa ca e4 d3 b0 :<f2>
>1460 0b 20 e1 11 68 68 20 79 :<e3>
>1468 04 4c 4d 8d 20 73 04 d0 :<72>
>1470 fb 60 a2 00 86 15 e8 86 :<01>
>1478 14 78 e8 8e 0a ff a9 cd :<a4>
>1480 8d 0b ff 4c c7 12 4c 1c :<e6>
>1488 99 20 d8 9d e0 26 b0 f6 :<fc>
>1490 86 de 20 d8 9d e0 17 b0 :<21>
>1498 ed 86 db 20 d8 9d ca ca :<24>
>14a0 e4 de 90 e2 e8 e8 86 dc :<55>
>14a8 20 d8 9d ca ca e4 db 90 :<9d>
>14b0 d5 e8 e8 86 dd 20 d8 9d :<44>
>14b8 e0 03 b0 ca 8a 0a 0a 0a :<dd>
>14c0 a8 a2 00 b9 58 15 9d 5e :<f7>
>14c8 06 c8 e8 e0 08 90 f4 20 :<5d>
>14d0 72 14 a6 db 20 20 12 a4 :<1d>
>14d8 de ad 65 06 91 d0 c8 ad :<0e>
>14e0 5e 06 91 d0 c8 c4 dc 90 :<b8>
>14e8 f9 ad 5f 06 91 d0 a6 dd :<01>
>14f0 20 20 12 a4 de ad 63 06 :<ca>
>14f8 91 d0 c8 ad 62 06 91 d0 :<35>
>1500 c8 c4 dc 90 f9 ad 61 06 :<d1>
>1508 91 d0 a6 db e8 20 20 12 :<60>
>1510 ad 64 06 a4 de 91 d0 ad :<6a>
>1518 60 06 a4 dc 91 d0 e8 e4 :<0d>
>1520 dd 90 ea 20 20 12 a4 de :<48>
>1528 ad 63 06 91 d0 c8 ad 62 :<65>
>1530 06 91 d0 c8 c4 dc 90 f9 :<42>
>1538 ad 61 06 91 d0 a6 db e8 :<76>
>1540 8e e6 07 a6 de e8 8e e7 :<2a>
>1548 07 a6 dc ca 8e e8 07 a6 :<aa>
>1550 dd ca 8e e5 07 4c 9a d8 :<d3>
>1558 41 42 44 47 46 45 43 40 :<c4>
>1560 48 49 4a 47 46 45 43 40 :<c1>
>1568 4c 4d 4f 52 51 50 4e 4b :<4b>
>1570 00 1f 2f 27 23 20 21 21 :<b8>
>1578 00 ff ff ff ff 00 ff ff :<26>
>1580 00 fe fa f6 ee 1e de de :<e2>
>1588 21 21 21 21 21 21 21 21 :<05>
>1590 de de de de de de de de :<36>
>1598 21 20 22 25 2a 15 00 00 :<9e>
>15a0 ff 00 aa 55 aa 55 00 00 :<db>
>15a8 de 1e ae 56 aa 54 00 00 :<1f>
>15b0 00 ff ff ff ff 00 ff 00 :<18>
>15b8 00 fc fa f6 ee 1e de 5e :<6f>
>15c0 5e 5e 5e 5e 5e 5e 5e 5e :<6c>
>15c8 00 3f 3f 30 30 33 32 32 :<70>
>15d0 00 ff ff 00 00 ff 00 00 :<90>
>15d8 00 fc fc 0c 0c cc 4c 4c :<82>
>15e0 32 32 32 32 32 32 32 32 :<b2>
>15e8 4c 4c 4c 4c 4c 4c 4c 4c :<3d>
>15f0 32 33 30 30 3f 3f 00 00 :<9d>
>15f8 00 ff 00 00 ff ff 00 00 :<d3>
>1600 4c cc 0c 0c fc fc 00 00 :<c9>
>1608 01 03 1d 3e 36 3e 1d 63 :<97>
>1610 fe cf b7 b7 cf f3 f3 ff :<fb>
>1618 ff ab d5 ff fc 80 9f ff :<1d>
>1620 ff ff 83 ff 83 ff 83 ff :<cd>
>1628 00 00 0d 1e 1e 0d 13 3f :<f2>
>1630 00 fe e6 e6 fe e6 e6 de :<3e>
>1638 3f 3f 20 39 20 3f 00 00 :<68>
>1640 be fe 02 ce 02 fe 00 00 :<a6>
>1648 00 00 00 00 0d 0d 03 0f :<61>
>1650 00 00 00 f8 f8 f8 b8 78 :<3f>
>1658 1c 1c 1f 10 1f 00 00 00 :<09>
>1660 c8 c8 f8 08 f8 00 00 00 :<ce>
>1668 00 00 3f 30 32 35 34 30 :<10>
>1670 00 00 fc 0c 4c 4c ac 0c :<86>
>1678 3f 7f 61 7f 15 2a 7f 00 :<31>
>1680 fc fe 86 fe 58 ac fe 00 :<83>
>1688 7f 43 49 43 49 49 7f 00 :<b7>
>1690 7f 41 4f 43 4f 4f 7f 00 :<ea>
>1698 18 3c 3c 18 18 ff ff 18 :<b6>
>16a0 18 3c 3c 66 66 c3 c3 00 :<73>
>16a8 00 ff c0 00 00 ff c0 0e :<3b>
>16b0 00 ff 00 00 70 ff 70 70 :<df>
```

```
>16b8 00 ff 00 1c 1c ff 1c 1c :<1d>
>16c0 0e ff ce 0e 0e ff ca 00 :<7e>
>16c8 73 ff 73 73 73 ff aa 00 :<03>
>16d0 9c ff 9c 9c 9c ff aa 00 :<b5>
>16d8 60 60 60 60 60 60 68 6c :<6f>
>16e0 00 00 00 00 00 00 42 63 :<99>
>16e8 00 00 00 00 00 00 10 18 :<f7>
>16f0 6e 6f 6f 7f 79 6f 6f 6f :<44>
>16f8 73 7b ff ff ce ff cf cf :<cb>
>1700 9c de fe fe 72 fe fe fe :<bc>
>1708 00 00 00 00 00 00 00 ff :<a6>
>1710 00 00 00 00 00 00 ff ff :<76>
>1718 00 00 00 00 00 ff ff ff :<66>
>1720 00 00 00 00 ff ff ff ff :<66>
>1728 00 00 00 ff ff ff ff ff :<6e>
>1730 00 00 ff ff ff ff ff ff :<7a>
>1738 00 ff ff ff ff ff ff ff :<88>
>1740 ff fe fc f8 f0 e0 c0 80 :<6d>
>1748 00 00 00 ff ff 00 00 00 :<8f>
=====================================
P R O G R A M M E N D E
=====================================
```

Hardcopy

Mit einer kleinen Maschinenroutine ist das Ausdrucken einer Hardcopy vom Grafikbildschirm keine Schwierigkeit mehr. Nachdem Sie das Hardcopy-Programm geladen und gestartet haben, können Sie entweder eine Grafik entwerfen oder ein Programm mit HIRES-Grafik starten. Ein SYS-Aufruf setzt den Ausdruck in Gang.

Eine Hardcopy vom Grafikbildschirm war der Wunsch vieler Leser. Nun ist sie da. Die Routine ist in Maschinensprache geschrieben, damit die Druckerausgabe mit annehmbarer Geschwindigkeit vonstatten gehen kann. Ein weiterer Vorteil gegenüber BASIC besteht darin, daß nichts von dem ohnehin geringen Speicherplatz des C16/116 verlorengeht, verbleiben doch im Grafikmodus nur mehr etwa zwei KByte für BASIC-Programme.

In der Folge werden wir auch Hardcopies für weitere Druckertypen veröffentlichen und die Bedienung etwas komfortabler gestalten, nämlich Ausdruck auf Knopfdruck. In der vorliegenden Fassung wird, noch etwas umständlicher, mit einem SYS-Aufruf hantiert. Wenn Sie diesen Aufruf nicht bereits in Ihrem Programm vorsehen, müssen Sie es mit Hilfe der STOP-Taste unterbrechen.

Betätigen Sie bei gedrückter STOP-Taste zusätzlich den Resetknopf, landen Sie im Monitor, den Sie wieder mit X und RETURN verlassen können. Dieses Verfahren hat den Vorteil, daß Sie nicht blindlings versuchen müssen, die Umschaltung in den Textmodus oder den SYS-Aufruf ohne Sichtkontrolle einzugeben. Im Textmodus, in den Sie auf diese Weise sicher gelangen, geht dies nämlich viel besser.

Da der Inhalt des Grafikbildschirmes nicht verlorengeht, läßt er sich, selbst, wenn auf dem Textbildschirm etwas anderes zu sehen ist, noch ausdrucken. Nach einem SYS 1525 können Sie das Ergebnis bewundern. □

HIRES-HARDCOPY

```
10 rem hires hardcopy==========c16 <pi>
20 rem (p) comodore welt team      <co>
30 rem =========================== <ng>
40 rem (c) by alfons mittelmeyer   <cg>
50 rem                             <pd>
60 rem c16/116/plus4               <kb>
70 rem und commodore-drucker       <ma>
80 rem aufruf mit sys 1630         <po>
90 rem =========================== <jg>
100 fori=1015to1076                <dh>
110 reada:pokei,a:next             <ne>
120 fori=1630to1777                <hd>
130 reada:pokei,a:next             <gp>
140 data 138,072,198,216,208,017   <dl>
150 data 169,056,024,101,214,133   <ch>
160 data 214,169,001,101,215,133   <kj>
170 data 215,162,008,134,216,230   <dj>
180 data 214,208,002,230,215,104   <ko>
190 data 170,096,162,002,181,208   <hk>
200 data 149,211,202,016,249,096   <pk>
210 data 162,002,181,211,149,214   <mj>
220 data 202,016,249,096,162,002   <al>
230 data 181,214,149,208,202,016   <ag>
240 data 249,096                   <id>
250 data 169,004,162,004,160,255   <bh>
260 data 032,186,255,032,192,255   <lf>
270 data 162,004,032,201,255,169   <ne>
280 data 008,032,210,255,169,255   <ap>
290 data 133,219,169,032,133,209   <lg>
300 data 169,000,133,208,169,008   <pi>
310 data 133,210,169,228,133,217   <fg>
320 data 032,023,004,032,033,004   <nj>
330 data 160,007,032,247,003,136   <ll>
340 data 208,250,032,043,004,032   <fo>
350 data 033,004,169,040,133,218   <io>
360 data 162,000,160,000,177,214   <lf>
370 data 149,220,032,247,003,232   <ik>
380 data 224,007,208,242,160,008   <pj>
390 data 162,007,022,220,042,202   <fo>
400 data 016,250,009,128,037,219   <me>
410 data 032,210,255,136,208,238   <ld>
420 data 165,211,024,105,008,133   <nl>
430 data 211,144,002,230,212,032   <ec>
440 data 033,004,198,218,208,202   <cb>
450 data 169,013,032,210,255,230   <oc>
460 data 217,048,169,169,143,133   <bl>
470 data 219,165,217,240,161,169   <ma>
480 data 015,032,210,255,032,204   <bk>
490 data 255,076,195,255           <no>
500 rem ========================== <mp>
510 rem  p r o g r a m m e n d e  <hf>
520 rem ========================== <cm>
```

Ist Ihr Computer auch schon krank?

Viren unbekannter Herkunft nagen an Ihrem wertvollen Programmbestand. Die wildesten Gerüchte sind im Umlauf. Ist die millionenfache Verbreitung noch aufzuhalten, sind schon Mittel und Wege zur Abwehr dieser Computerschädlinge gefunden? Hier einige Tips & Tricks zum Vorbeugen und Verjagen.

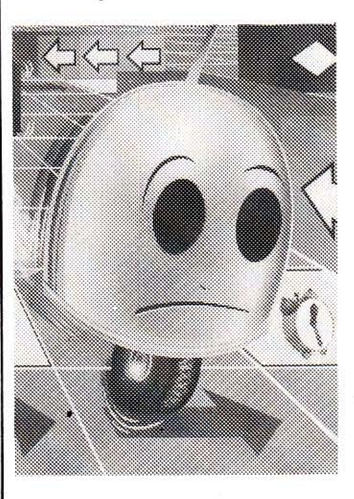
Laufend berichten die verschiedenen Medien vom Auftauchen unterschiedlich harmloser oder gefährlicher Computerviren. Da dieses Programmierkunststückchen gedankenloser Hacker immer mehr um sich greift, wollen wir Ihnen ein paar Tips geben, wie Sie Ihre Daten vor der Zerstörung schützen können.
Ein Computervirus ist ein sich selbst verbreitendes Programm, analog dem biologischen Virus ist es ohne einen Wirt nicht fähig, sich zu vermehren. Es benötigt dazu einen Computer und ein anderes Programm.

BHP-VIRUS BESONDERS LÄSTIG

Der von der „Bayrischen Hackerpost" entwickelte und verbreitete BHP-Virus, der unter den C64-Usern grassiert, ist vergleichsweise harmlos, deshalb aber nicht weniger lästig. Er begnügt sich damit, sich zu vermehren, ohne die Programme direkt zu zerstören. Er wird für 10 Mark verkauft oder unter Computer-Freunden spaßeshalber weitergegeben.
Als „Medikament" gegen seine Verbreitung wird von verschiedenen Anbietern bereits ein Virenkiller-Programm angeboten, das verseuchte Disketten vom Virus befreien soll. Ein solches Programm kann natürlich nur einen ganz bestimmten Schädling erkennen oder beseitigen. Bereits kleine Änderungen im Virus-Programm machen den Killer wieder zahn- und wirkungslos.

Wie funktioniert ein Computervirus?

Damit er sich vermehren kann, muß der Virus sich auf der Diskette vor einem Programm festsetzen. Dabei wird das Disketten-Inhaltsverzeichnis manipuliert, damit nicht gleich auffällt, daß das Programm verseucht ist. Beim BHP-Virus ist ein



Das Wort Viren treibt so manchen Computeranwender den Angstschweiß auf die Stirn. Denn diese kleinen Lebewesen treiben es ganz schön bunt mit so manchem Programmbestand.

Programm durch einfaches Listen als infiziert zu erkennen. Die erste Zeile lautet nämlich:

```
1986 sys peek(43)+peek
(44)*256+48:virus
```

Erst nach dem Starten des Programms wird es gefährlich. Jetzt verschwindet der Virus in den Untergrund und zwar unter den I/O-Bereich ab $D000. Gleichzeitig wird er RESET-fest gemacht und verändert einige Systemvektoren. Dadurch wird er beim Laden eines Programmes aktiv.
Das Verseuchen einer Diskette geschieht also nicht, wie oft vermutet, beim SAVE-Kommando, sondern beim Laden eines beliebigen Programms. Auf diese Weise fällt die Aktivität des Virus-Programms erst dann auf, wenn es bereits zu spät ist.

Schäden durch Viren-Programme

Was kann ein solches Programm anrichten? Die meisten harmloseren verbrauchen nur Speicherplatz. Der BHP-Virus zum Beispiel benötigt jedesmal, wenn er sich vor ein Programm setzt, etwa 8 KByte.
Bei häufigem Gebrauch von infizierten Disketten kommt ein enormer Speicherplatzverbrauch zustande. Es gibt jedoch wesentlich gefährlichere Viren-Programme.

FESTPLATTEN STARK GEFÄHRDET

Diese können auf Festplatten oder gar in großen Datenbanken unersetzliche Daten vernichten. Spätestens hier hört der Spaß auf, und die Kriminalität beginnt.
Besonders gefährdet sind die teueren und benutzerfreundlichen Computer, die ein im Boot-Sektor gekennzeichnetes Programm beim Einschalten automatisch laden. Hier hat ein Virus-Programmierer ideale Hardware-Voraussetzungen, um einen Virus in das System einzuschleusen. Das gleiche gilt für Autostart-Programme.

Erkennen von infizierten Disketten

Da ein Virus Platz benötigt, muß er sich irgendwie feststellen lassen. Dafür gibt es prinzipiell zwei Möglichkeiten:

1. Das infizierte Programm erscheint im Directory um den Virus verlängert.
2. Das Directory wurde verändert, so daß die angezeigte Länge mit der tatsächlichen originalen Programmlänge nicht mehr übereinstimmt.

Im ersten Fall muß die tatsächliche Programmlänge bekannt sein, um die Veränderung zu bemerken. Im zweiten Fall hilft nur ein VALIDATE und rechnen.
Stimmt nach dem VALIDATE die Summe aus der Anzahl an freien und belegten Blöcken nicht mit den möglichen freien Blöcken nach dem Formatieren überein, so liegt der Verdacht nahe, daß ein Computer-Virus zugeschlagen hat.

Hygiene ist der beste Schutz

Wer nur eigene Programme verwendet, braucht sich über Computerviren keine Gedanken zu machen. Besonders gefährlich sind hingegen Public-Domain-Programme und Raubkopien. Es kursieren Gerüchte, wonach sogar Profi-Programmierer, die sich darüber ärgerten, daß mehr Schwarzkopien als Original-Programme im Umlauf sind, selbst infizierte Raubkopien ver-

breitet haben, um den Schwarzkopierern eines auszuwischen. Wer fremde Software verwendet, sollte einige Sicherheitsregeln beachten:

1. Fremde Disketten untersuchen. Laden Sie ein Programm nach dem anderen und listen Sie es, um verräterische SYS-Befehle zu finden (ein Virus-Programm ist immer ein Maschinenprogramm). Diese Programme vorher nicht starten! Vorsicht vor Programmen, die nachladen.
2. Fremde und eigene Disketten stets trennen. Nicht wegen der Ansteckungsgefahr, sondern wegen des besseren Überblicks.
3. Nach Verwendung von fremden Disketten den Computer „desinfizieren", also ausschalten. Sicherheitshalber auch die Diskettenstation ausschalten, da es auch floppy-resistente Viren gibt.
4. Wenn möglich, machen Sie sich von wichtigen Programmen eine Sicherheitskopie auf Band.

Datasettenbenutzer sind vor Viren so gut wie sicher. Uns ist bisher kein Virus bekannt geworden, der auch bei Datasettenbetrieb wirksam wäre. Hier kann der Virus sich nicht unkontrolliert ausbreiten, da er sich nur beim Abspeichern des Programms, mit dem er eingeladen wurde, wieder mitabspeichern kann.
Sicherheitshalber sollten Sie bei verdächtigen Programmen den Computer immer ausschalten, bevor Sie andere Programme mit Ihrem System kopieren.
Also Finger weg von Disketten unbekannter Herkunft und undurchsichtigen Inhalts, es könnte ein Virus darauf sein. Safety first und keine Angst vor den bösen Hackern.

Rudolf Schmid-Fabian □

KOMPRESSOR CW SPEZIAL 5/88

Inzwiscnen ist es gelungen, kleinere Programme zu komprimieren. Bei größeren geht's nicht. Was muß man ändern, um zum Beispiel Mystika, ein sehr langes Programm, zu komprimieren?
Peter Bieger, Himmelpforten

Im BASIC-Kompressor kommen zwei Dimensions-Anweisungen vor. Damit durch leere Datenfelder nicht zuviel vom geringen Speicherplatz eines nicht erweiterten C16/116 verbraucht wird, wurde als maximale Zeilenzahl des zu komprimierenden Programmes der Wert 300 voreingestellt. Dies geschieht in Zeile 210 mit mz=300. Besitzt das BASIC-Programm eine höhere Zeilenanzahl, so braucht nur mz entsprechend angepaßt zu werden.

DEUTSCHE UMLAUTE IN DER PLUS4-DATEIVERWALTUNG

... Jetzt habe ich zwar auch hier Umlaute, aber dafür neue Probleme: Wenn ich mir Etiketten ausdrucken lasse, haut mir der Drucker (MPS 1000) manchmal Leerzeilen dazwischen, die ich nicht eingegeben habe. Diese Leerzeilen werden in unregelmäßigen Abständen mitten in den Adreßfiles ausgegeben, was sich gerade bei Etiketten sehr ungünstig auswirkt. Am Plus4 kann es nicht liegen, da ich meinen zur Zeit in Reparatur habe, und der, den ich mir geliehen habe, auch Leerzeilen bringt. Liegt es am Drucker oder am Programm? Als ich die Umlaute noch nicht verwendete, ist mir dieser Fehler noch nicht aufgefallen.
Michael Weidmann, Berlin

Es freut uns zu hören, daß unser Zeichensatzprogramm auch mit einem Commodore-Drucker, der einen deutschen Zeichensatz besitzt, sogar in Verbindung mit der Plus4-Dateiverwaltung funktioniert. Daß dadurch unmotivierte Leerzeilen zustandekommen, konnten wir allerdings nicht beobachten.
Beim Nachvollziehen des Etikettenausdrucks stießen wir allerdings auf Ungenauigkeiten im Plus4-Software-Handbuch. Im Text auf Seite 210 wird geraten, die NEXTPAGE-Anweisung gleich im Anschluß an die PAPERSIZE-Einstellung vorzunehmen. Auf dem Bildschirmabdruck ist diese jedoch erst in der letzten Formatzeile zu finden. Unten auf Seite 211 heißt es gar, daß die beiden NEXTPAGE-Anweisungen nicht fehlen dürften. Ob deutscher Zeichensatz oder nicht, bei NEXTPAGE nur am Anfang trat nach einer gewissen Anzahl von Zeilen der von Ihnen geschilderte Fehler auf, bei NEXTPAGE am Ende dagegen nicht, oder war es etwa umgekehrt? Eine der drei angesprochenen Versionen führt bestimmt zum gewünschten Ergebnis.

FEHLER BEI READ-ANWEISUNGEN

Ich habe den Checksummer aus Heft 10/87 eingegeben. Dieser läuft jedoch nicht, obwohl kein Eingabefehler vorliegt. Der C16 steigt immer aus bei der Zeile
240 fori=312 to 386 :readx:pokei,x:next
mit Fehlermeldung 14: illegal quantity error in Zeile 240.
Können Sie mir mitteilen, warum, und was ich ändern muß?
Harald Banghard, Laufenburg-Binzgen

Mit einer solchen Fehlermeldung mag schon mancher schwer zu kämpfen gehabt haben. Auch wer eine solche Zeile tausendmal überprüft, vermag keinen Fehler zu entdekken, denn der ist ganz woanders zu suchen: Wenn die READ-Anweisung beim Lesen von Datas, statt auf eine Zahl, unversehens auf einen Text stößt, ist es schon geschehen. Es fragt sich nur, wo.
Ein kleiner Trick hilft oft weiter: Wollen wir beobachten, wo das Programm abstürzt, können wir dies durch eine PRINT-Anweisung erreichen, die wir nach dem READ-Befehl einfügen. Unsere ursprüngliche Zeile wandeln wir am besten so ab:

```
240 fori=312 to 386:
    readx:printx,:pokei,
    x:next
```

Die eingelesenen Daten werden nun auf den Bildschirm ausgegeben. Das dem zuletzt sichtbaren Zeichen folgende DATA ist der Schwarze Peter. Der braucht aber noch lange nicht an allem schuld zu sein. Eventuell haben wir bereits vorher durch fälschliche Auslassungen oder Hinzufügungen dafür gesorgt, daß der READ-Befehl nun auf etwas stößt, das zu diesem Zeitpunkt gar nicht für ihn vorgesehen war. In diesem Falle kann uns nur noch der Vergleich der DATAs auf dem Bildschirm mit denen im Heft einen Aufschluß geben.

Haben Sie Fragen?
Commodore-Welt
Postfach 1161
D-8044 Lohof

Bastard - echt kriminell

Die meisten Computerfreaks beschäftigen sich neben der Lektüre von Fachmagazinen und Büchern bisweilen auch mit etwas leichterer Kost in Form von Filmen, die sich mit dem Computer beschäftigen. Da es in solchen Produkten meist mit erheblichem Hang zu Klischees und übertriebenen Utopien zugeht, nahmen wir einmal – zur Information für Sie – die Produktion eines Filmes mit dem Schwerpunkt Computer unter die Lupe.

Filme wie Wargames der L.I.S.A. dürften den meisten unserer Leser bekannt sein. Hauptgegenstand dieser Spielfilme war – wie auch in anderen Werken – der Computer. Doch zumeist neigen solche Streifen zu übertriebenen Utopien

MEHR SCIENCE-FICTION ALS REALITÄT

und tendieren mehr zur Science-fiction als zur Realität.
Um so mehr wurde unser Interesse von einer Presseinformation aus den Bavaria Filmstudios geweckt. Quasi vor unserer Haustür fanden die Dreharbeiten zu einer sechsteiligen Kriminalserie für das deutsche Fernsehen statt, deren Hauptgegenstand das Thema Computerkriminalität ist. Wir wollten hinter die Kulissen sehen und feststellen, wie solche Filme überhaupt zustande kommen. In Gesprächen mit Schauspielern und Produzent sowie der Besichtigung von Kulisse und einem der Drehorte informierten wir uns vor Ort.
Die Serie wird voraussichtlich Ende des Jahres unter dem Titel Bastard im deutschen Fernsehen ausgestrahlt. Es ist die Geschichte eines Computerfreaks, der sich auf eine Baleareninsel zurückgezogen hat und dort Spiele programmiert. Auf seinen nächtlichen Reisen

VERBRECHER-ORGANISATION MIT DATENBANK

in internationalen Datennetzen verfängt er sich in den Maschen einer Datenbank, die offensichtlich von einer weltweit operierenden Verbrecherbande benutzt wird. Solch böse Bits bringen den Herrn wieder ins Alltagsleben zurück, und er widmet sich der Verbrechensbekämpfung. Um diesen Anfang rankt sich dann die gesamte Handlung.
Das Grundproblem eines solchen Filmes sieht Produzent Peter Bradatsch darin, daß zum einen die Technik zwar wahrheitsgetreu, zum anderen aber möglichst effektvoll in

UND PIEPEN MUSS ER AUCH

Szene gesetzt werden soll. So sollen Zuschauer, die mit dem Medium Computer vertraut sind, keine unmöglichen Techno-Märchen aufgetischt bekommen. Andererseits müssen aber von der Materie Unberührte von der wundersamen Computerwelt gefesselt werden. Auf der einen Seite also die nüchterne Realität eines ganz normalen Personal Computers, auf der anderen einen blinkendes und piependes Etwas, welches mit der eigentlichen Realität nichts mehr zu tun hat.
Um hierbei nicht allzu sehr auf das Niveau von Filmen wie Wargames zu sinken, bediente man sich bei der Produktion von Bastard eines Programmierers, welcher durch Bera-





Oben: Der Darsteller der Titelrolle, ein echter Bastard, wird von seinen Kollegen verwöhnt.

tung und Entwicklung geeigneter Programme möglichst sämtliche Belange unter einen Hut bringen sollte. Bradatsch meint, so das Problem in den Griff bekommen und eine gewisse Ausgewogenheit erreicht zu haben. Der Kompromiß besteht beispielsweise in einem realitätstreuen Terminalprogramm, welches aber dem besseren Effekt zu-

BERATUNG DURCH PROGRAMMIERER

liebe mit überdimensionaler Schriftgröße und piependem Bildschirmaufbau versehen wurde. Vom breiten Publikum wird ein Computer anscheinend nur dann als ein solcher erkannt, wenn er kräftig Laut gibt. Freaks mögen es dem deutschen Fernsehen verzeihen. Interessiert hat uns auch ein weiterer Aspekt: Wie sieht es mit Unterstützung

UNTERSTÜTZUNG VON HERSTELLERN

seitens der Computerhersteller aus? Wohl jeder assoziiert inzwischen das neumodische Wörtchen Product Placement, wenn beispielsweise Schimanski im Tatort sehr ausführlich ein Hustenbonbon zu Munde führt. Ähnlich verhält es sich auch im Bereich der Geräte-Ausstattung, wobei Sparmaßnahmen allerdings im Vordergrund stehen und die daraus resultierende Werbung in Kauf genommen wird. Schließlich wäre auch der Zuschauer, der letzten Endes die Requisiten mit seinen Gebühren bezahlt,

SPARMASSNAHMEN IM SINNE DES ZUSCHAUERS

unzufrieden, wenn bei den Produktionen ein sechsstelliger Betrag für technische Kulissen ausgegeben würde. Auch bei Bastard war es so: Man wandte sich an drei große Hersteller – welche, werden Sie im Herbst selbst sehen – von PCs, die prompt zusagten und die Produktion mit der notwendigen Hardware versorgten.
Darüber hinaus gab es aber auch noch Rat und Hilfe vor Ort. So sorgte beispielsweise die Bildwiederhol-Frequenz von sechzig Hertz eines PC für Probleme bei den Aufnahmen: Ein unschöner Streifen wanderte über den Schirm. Jeder, der schon einmal versucht hat, ein Bildschirmfoto zu machen, kennt die Balken, die entstehen, wenn die Belichtungszeit zu kurz ist. Und Filmen ist schließlich nichts anderes als Fotografie in Serie. Um dies zu beheben, wurden flugs einige Techniker der betreffenden Firma eingesetzt, um den Rechner mit einer anderen Grafikkarte auszurü-

TECHNISCHE PROBLEME MIT GRAFIKKARTE

sten. Gratis und mit Freuden, versteht sich, denn die Firma hatte schließlich kein Interesse an wandernden Bildern.
Die Sparsamkeit in der Requisite ermöglichte vielleicht die ausgezeichnete Besetzung des Filmes. Peter Sattmann spielt die Hauptrolle, jenen ausgestiegenen Programmierer und DFÜ-Freak. Für ihn ist der Computer nichts Neues. Er sei schon seit Jahren

ECHTE STARBESETZUNG

durch mehrere private Rechner damit vertraut, erklärt er. Aus der Sammlung hat er den Apple zum Lieblingsspielzeug erkoren.
Ein Bonbon für die Fotografen war natürlich Gudrun Landgrebe. Als Lisa Pallmann darf sie dem Kämpfer um Computer-Gerechtigkeit zur Seite stehen. Weitere Rollen wurden mit Heinz Reincke und Dieter Krebs besetzt.
Zusammen mit dem Regisseur Ulrich Stark geht also keine unbekannte Truppe ans Werk. Mit einer Ausnahme allerdings: dem Bastard. Das Hunde-Viecherl, das der Serie den Namen gab, nennt sich im Film Diddle und heißt in Wirklichkeit Hexie. Es ist eben alles nur Schein beim Film, doch auch hier wieder mit einer Ausnahme: der Bastard ist tatsächlich einer. *TS* □





Oben: Gudrun Landgrebe und Peter Sattmann. Links: Alle Darsteller der Fernsehserie

Nobody is perfect

Der Teufel steckt im Detail

Niemand ist gegen Fehler gefeit, die sich hartnäckig in den Programmen festsetzen und auch Anwender mit ruhiger Natur zur Explosion bringen. Hier hilft nur stetige Fehlersuche, auch in einem „fertigen" Programm.

Rüdiger Valk, Informatikprofessor an der Universität Hamburg, befaßte sich in der Süddeutschen Zeitung mit der Frage nach der Leistungsfähigkeit der Computer und ihrer Programme und untersuchte die Vertrauenswürdigkeit der Systeme. Sein Fazit: Sie sind „schnell, genau und fehlbar".
Es waren zwar damit Großrechneranlagen gemeint, die einem Einpersonenhaushalt eine 13 000-Mark-Stromrechnung schicken oder von einem Dreijährigen den Führerschein (für seinen Tretroller?), einziehen wollen, oder einen 80-jährigen zur Bundeswehr berufen. Dies ist auf den ersten Blick vielleicht noch lustig, aber wenn Computer mit ihrer Fehlbarkeit Menschenleben gefährden, wie bei einem Flugzeug oder gar einem Atomkraftwerk, dann spätestens hört der Spaß auf. Aber ohne rechnerunterstützte Systeme sind viele Bereiche des täglichen Lebens längst nicht mehr denkbar.

FEHLERFREIE PROGRAMME MÖGLICH?

Die in der COMMODORE WELT und COMMODORE WELT SPEZIAL abgedruckten Programme gefährden hoffentlich niemanden, aber es ist trotzdem lästig oder gar ärgerlich, wenn die mühevoll eingetippten Programme nicht das gewünschte Ergebnis erzielen. Auch für uns, mit Verlaub, gilt: Nobody is perfect. Auf dem langen Weg vom Programmautor, Redakteur, drucktechnischen Hersteller, bis zu Ihnen, dem Leser, sind einige Fallstricke eingebaut, über die auch wir manchmal stolpern. Trotz vieler Sicherheitsvorkehrungen rutschen auch den Korrektoren Fehler durch die Finger. Zum Beispiel bei einem Spiel, bei dem in der 99. Spielstufe, wenn die Konstellation Weihnachten auf Ostern fällt, ein Hund den Radfahrer in Peking beißt und dieser vom Pink Panther verhaftet wird, – dann passiert ein glatter Ausstieg ins RAM/ROM-Nirwana, dann ist jeder wie auch immer geartete Test überfordert, wenn diese Fehler in einer Wahrscheinlichkeits-Konstellation von nicht nennbarer Größe auftritt.

FEHLERQUOTE: 0,02 PROMILLE

Ein 50-KByte-Programm, das die Eingabe von mehr als 500 000 Zeichen erfordert, ist nach Eingabe ohne Kontrolle und Korrektur garantiert nicht lauffähig. Denn nur 0,01 Prozent Fehler, oder gar 0,02 Promille Fehler, das sind fünf beziehungsweise ein möglicher Tippfehler, können das Programm zum Absturz bringen. Mit Eingabehilfen wie Checksummer ist die Fehlerquote fast auf Null zu reduzieren, das heißt natürlich nur, wenn das Originalprogramm in Ordnung ist.
Hier eine kleine Zusammenstellung einiger Programme, die in der COMMODORE WELT oder in einem SPEZIAL erschienen sind und bei denen der Computer-Fehlerteufel zugeschlagen hatte.

COMMODORE WELT SPEZIAL 1/88 Seite 20

TEXT C16

```
540 printb2$"PLZ"
b2$"Ort"b4$b4$;:
inputc$                <gj>
```

Alle Zeilen, die eine PRINT#-Anweisung enthalten, sind wichtig für Steuerung des Druckers und müssen auf die Codes eingestellt sein, die der angeschlossene Drucker auch versteht. Manche brauchen zur Erkennung von Steuercodes immer ein vorangestelltes ESCAPE. Dies ist der Code CHR$(27), vor den folgenden Befehlscode, um druckbare Zeichen von Steuercodes zu unterscheiden (siehe Tabelle 1).

COMMODORE WELT SPEZIAL 5/88 Seite 61

MYSTIKA ADVENTURE

```
2660 ifri<>0orog<>0
thenan$="wie bitte !!"
:goto190               <oj>
```

COMMODORE WELT SPEZIAL 5/88 Seite134

MENSCH ÄRGERE DICH NICHT

```
2380 printchr$(18)"dich
nicht... "             <cc>
```

Unbedingt das letzte Space nach den drei Punkten nicht vergessen, da sonst am Bildschirm die ganze Grafik verschoben wird. Die Zeile ist im abgedruckten Programm zwar richtig, aber auf Grund vieler Anrufe haben wir hier eine mögliche Fehlerquelle entdeckt.

COMMODORE WELT SPEZIAL 5/88 Seite 48

WOLFSOFT-DATEI

```
3120 x$=z3$:m=35:
gosub3880:printx$  <dj>
```

Es wurde eine falsche Sprungadresse benutzt, die zu einem Verschieben der Bildschirmausgabe führte.

COMMODORE WELT SPEZIAL 4/87 Seite90

JOHN BEND DETEKTIV

```
2570 pokezn,12:pokezs,
     22:sysx
2580 pokezn,11:pokezs,
     30:sysx
2625 print"Abenteuer."
     :printec$:end
```

Nach erfolgreicher Erledigung des Auftrages und der Belobigung durch Mr. Q ist das Game in Zeile 2625 zu Ende. Dies wurde in der abgedruckten P4-Version vergessen. Die Zeilen 2570/2580 verhindern jetzt das Verschieben der Grafik bei einer – jetzt richtigen – Zeile/Spalte Angabe.

COMMODORE WELT SPEZIAL 3/88 Seite135

BOOT

```
130 char,0,z,d$:poke239,
```

TABELLE

Zeile		Code		Funktion	
		ec$	=	String für ESCAPE CHR$(27)	
Zeile 390		chr$(15)	=	Schmalschrift ein	(Compressed)
Zeile 400	ec$	chr$(52)	=	Kursivschrift ein	(Italic im High Memory)
Zeile 410	ec$	chr$(88);			
		chr$(01)	=	Schönschrift ein	(NLQ)
Zeile 610		chr$(14)	=	Breitschrift ein	(Enlarged)
Zeile 620	ec$	chr$(87);			
		chr$(00)	=	Breitschrift aus	(Enlarged)
Zeile 630	ec$	chr$(80)	=	Elite aus PICA ein	
Zeile 640	ec$	chr$(69)	=	Fettdruck ein	(Enhanced)
Zeile 750	ec$	chr$(70)	=	Fettdruck aus	(Enhanced)
Zeile 660		chr$(20)	=	Breitschrift aus	(Enlarged)
Zeile 680	ec$	chr$(65);			
		chr$(n)	=	Zeilenabstand auf n/72 Zoll	n = 0 bis 85
Zeile 690	ec$	chr$(45);			
		chr$(01)	=	Unterstreichen ein	(Underlining)
Zeile 760	ec$	chr$(83);			
		chr$(01)	=	Tiefstellen ein	(Subscript)
Zeile 780	ec$	chr$(84)	=	Tiefstellen aus	(Subscript)
Zeile 790		chr$(18)	=	Schmalschrift aus	(Compressed)
Zeile 800	ec$	chr$(74);			
		chr$(n)	=	Papiertransport n/216 Zoll	n = 0 bis 255

Angaben für Mannesmann Tally 85

```
7:t=1319
140 poket,145:poket+1,
145:poket+2,13:poket+3,
147:poket+4,82:poket+5,
213:poket+6,13
```

Die verwendeten Codes in Zeile 140 sind:
145 CURSOR UP
13 RETURN
147 SCNCLR
82 Zeichen CHR$(82) R
213 Zeichen CHR$(213) SHIFT U

COMMODORE WELT SPEZIAL 2/88 Seite 48

UNI-DATEI II

Korrektur 1

```
99 fast:fori=.to9:a$=a$
+z1$+z1$+z1$+z1$+z1$
+z1$+z1$+z1$:nexti:
md%=20:dima(md%),a$
(md%):window.,.,79,
24                      <ma>
100 bank15:sys4924:
sys4864:printcl$c4$rn
$bl$a$:sys52332,,22,
:printa$.f%=8:f$="0"
:g%=4:gosub1610:color
6,1:goto1660            <nn>
1000 ifa$=x1$anda<na
-9goto1070:elseifa$=
ec$anda>10goto1100
                        <hh>
1740 sys4888,ze,ze:b=
len(a$):printcl$x2$sl$
lg$tab((80-b)/2)a$wh$;
:goto1610               <fh>
60030 x1$=chr$(010)
:x2$=chr$(011)          <li>
```

Korrektur 2

```
440 printcl$:clr:dima$
(20),a(20):f%=8:f$="0"
:g%=4:restore1900:read
a$,ze:gosub1740         <bc>
```

Um einen einwandfreien Ablauf des Programmes zu erreichen, tippen Sie bitte die oben genannten Zeilen zum Listing dazu oder ändern die alten entsprechend ab. In Zeile 440 ist unbedingt der Befehl Bildschirm löschen (PRINT CL$) zu ergänzen,

COMMODORE WELT SPEZIAL 2/88 Seite 48

EISENBAHN 64

```
1610 ife(i)=2thena=(peek
     (56320)or128)
5355 -rc,9,40
6170 gosub3300
6171 fori=1to8
6172 fa(i,2)=0
6173 next
6174 -suy,1:-suc,5:-sue
6179 next
6180 -suf,0:-suc,1:-suc,
     32:poke198,0
```

Die Minuszeichen in den Zeilen 5355, 6174 und 6180 bedeuten den Pfeil nach links.

Die Programme Eisenbahn und Yatzee laufen nur wenn vorher das Programm RAND & SPRITE gestartet wurde. Bei der Eingabe der Programme ist darauf zu achten, daß zum Beispiel in den Zeilen 160/170/210/260 usw. bei Yatzee, anstatt des IBM-ASCII-Grafikzeichens CHR$(95), der Commodore-ASCII-Zeichencode CHR$(95) Pfeil-nach-Links eingegeben wird. Das selbe gilt für das Spiel Eisenbahn im selben Heft.

Robert Wagner □

IMPRESSUM

COMMODORE-SPECIAL

erscheint sechsmal jährlich in der CA-Verlags GmbH (i.G.)

VERANTWORTLICH FÜR DEN INHALT:
Alfons Mittelmeyer

REDAKTION UND STÄNDIGE MITARBEITER:
Peter Basch, Harald Beiler, Rosemarie Huber, Lothar Miedel, Michael Reppisch, Rudolf Schmid-Fabian, Torsten Seibt, Hermann Wellesen, Bernd Welte

GESCHÄFTSFÜHRER (und verantwortlich für Anzeigen):
Werner E. Seibt

ANSCHRIFT FÜR ALLE VERANTWORTLICHEN:
Postfach 1107,
8044 Unterschleißheim
Tel.: 089/129 80 11
Telex: 5214428 cav-d
Es gilt Preisliste N. 8 vom 1.1.1988
Media-Unterlagen bitte anfordern.

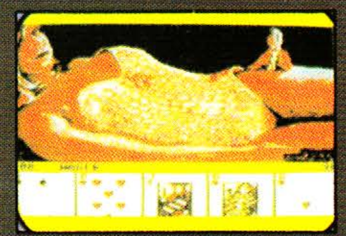
VERTRIEB:
Verlagsunion Wiesbaden

DIE NEUEN RENNER FÜR C-16 UND PLUS/4 SIND DA!









PLUS/4 SEXTETT

Die einzigartige Spielesammlung exklusiv für alle Plus/4-Besitzer (oder C-16/116 mit 64 K RAM). Nicht weniger als 6 hervorragende Spiele gibt's hier zum Preis von einem:

ALIEN INVASION
Räumen Sie in einem stark befestigten feindlichen Lager auf.

JUMP JET
Ein spannender Kampf- und Flugsimulator mit einem Senkrechtstarter.

KARATE KING
Das beste Karatespiel mit ausgezeichneter Animation durch flimmerfrei Multicolor-Sprites.

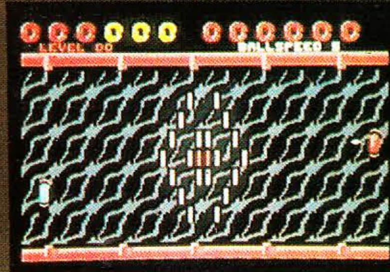
OUT ON A LIMB
Ein Kletter- und Springspiel nach dem bekannten Märchen.

STRIP POKER
Das „klassische" Kartenspiel mit dem gewissen Etwas; spielen Sie gegen Susi und Melissa.

TERRA NOVA
Vertikal scrollendes Actionspiel mit Sprachausgabe und toller, schneller Grafik.

Kassette oder Diskette **29.95**

DEMOLITION



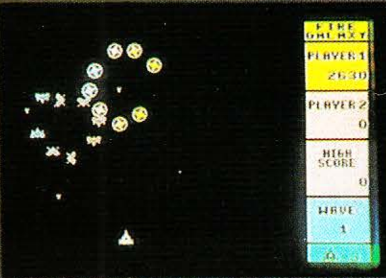
Die Neuauflage des Breakout-Klassikers sorgt mit 36 verschiedenen Leveln, 6 Schwierigkeitsstufen und 2-Spieler-Modus (gleichzeitig!) für reichlich Abwechslung.
Kassette oder Diskette **19.95**

FORTRESS UNDERGROUND



In einer gigantischen unterirdischen Höhle müssen Sie mit Ihrem Hubschrauber ein feindliches Kraftwerk aufspüren und eliminieren. Aber der Weg dorthin ist lang und gefährlich.
Kassette oder Diskette **19.95**

FIRE GALAXY



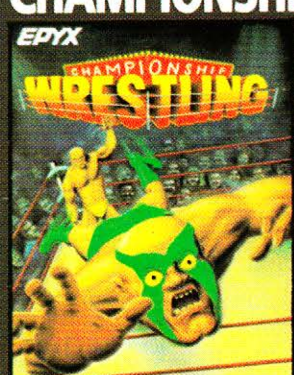
Das neueste Weltraum-Actionspiel von Henrik Wening: Immer neue Angreifer in immer neuen Formationen erwarten Sie. Hier darf nach Herzenslust geballert werden.
Kassette oder Diskette **9.95**

ACE 2



Der Nachfolger des berühmten Kampf- und Flugsimulators ACE ist da! Diesmal können 2 Spieler auf einem geteilten Bildschirm gleichzeitig gegeneinander antreten. Nur für PLUS/4 und C-16/116 mit 64 K RAM!
Kassette **26.95** Diskette **38.95**

CHAMPIONSHIP WRESTLING



Das berühmte Catcher-Spiel von EPYX gibt es jetzt auch in einer neuen Version für Ihren C-16 oder PLUS/4. Schaffen Sie es, Ihre furchterregenden Gegner „aufs Kreuz zu legen"?
Kassette **26.95**

SPY VS. SPY



Wer kennt sie nicht - die Abenteuer von „Spion gegen Spion"? Endlich können auch C-16- und PLUS/4-Besitzer an diesem Spielspaß teilhaben.
Kassette **26.95**

KINGSOFT-Coupon
☐ Bitte senden Sie mir Ihren aktuellen Katalog
Meine Adresse

Bitte fordern Sie heute noch mit nebenstehendem Coupon unseren neuen großen Gesamt-Katalog an mit einer Riesenauswahl an weiteren Produkten für Ihren C-16/116 oder PLUS/4.
Einfach Coupon ausfüllen, ausschneiden und ab die Post an
KINGSOFT, Grüner Weg 29, 5100 Aachen

WANTED
sucht
PROGRAMMIERER
für fast alle Computer-Typen, die gegen erstklassige Bezahlung Spiele von internationalem Niveau schreiben. Wenn Sie Interesse haben, wenden Sie sich noch heute an uns.

SPITZEN-SOFTWARE MADE IN GERMANY



GRÜNER WEG 29 · D-5100 AACHEN
☎ 0241/15 20 51 · Fax 0241/15 20 54